FAIRification, Quality Assessment, and Missingness Pattern Discovery for Spatiotemporal Photovoltaic Data

William C. Oltjen*, Yangxin Fan*, Jiqi Liu*, Liangyi Huang*, Xuanji Yu* Mengjie Li†, Hubert Seigneur‡, Xusheng Xiao*, Kristopher O. Davis†, Laura S. Bruckman*, Yinghui Wu*, Roger H. French*

*SDLE Research Center, Case Western Reserve University (CWRU), Cleveland, OH, 44106, USA

†University of Central Florida (UCF), Orlando, Florida, 32816, USA

‡Florida Solar Energy Center (FSEC), Cocoa, FL, 32922, USA

2022 IEEE 49th Photovoltaics Specialists Conference (PVSC) | 978-1-7281-6117-4/22/$31.00 ©2022 IEEE | DOI: 10.1109/PVSC48317.2022.9938523

***Abstract*—Due to the fast growth of the photovoltaic (PV) market, more power plants have become available with data accessible for power forecasting and long-term reliability assessment. The accuracy of the modeling on this data is influenced heavily by the quality of the data and can be improved through data imputation to fill missing gaps. In this study, we introduce a FAIRification framework for ingesting data from PV power plants. This process improves the efficiency of modeling on time series data provided by different labs and companies through an automated ingestion process. We take this analysis further by investigating the use of different imputation methods for filling in large chunks of missing data. Specifically, mean interpolation, linear interpolation, and k-nearest neighbors (KNN) were used in this report to fill in missing data for module temperature and power in a PV time series. It was found that the KNN algorithm outperforms the other methods due to its ability to leverage spatial coherence from nearby systems. These results point towards the potential use of a spatio-temporal graph neural network (st-GNN) in order to impute data using spatial coherence between systems in a large data set with time series data from many PV power plants.**

***Index Terms*—FAIRification, Spatiotemporal GNN, Missingness**

I. Introduction

Photovoltaics (PV) have become a dominant force in the energy sector over the past 20 years. The total, installed solar capacity has increased 500 times since 2000 to a total of 773 GW at the end of 2020 [1]. Not only has the field expanded so much in total, but the rate of installations continues to increase as well. In 2020, the world reported a new record of solar installations by implementing 138 GW of solar energy in a year [1]. The growth of the PV market has pushed the demand for power forecasting and performance evaluation for a huge population of PV power plants which have spatiotemporal coherence that can be utilized for improving model accuracy [2]. There are many logistical challenges towards performing this kind of time series analysis. Different groups use different types of databases, different variable naming schemes, different data cleaning processes, etc. The time series data itself is typically missing data, or can even have incorrect data from faulty sensors [3]. Manually addressing these kinds of issues takes time away from developing new models on the data which takes time away from producing more efficient PV modules based on the results. Through the development of an automated framework for ingesting time series data and feeding it into machine learning models, we can make our analysis methods more efficient at a larger scale. Much of the standards for our automated process are based on the FAIR principles introduced through the publication of Wilkinson et al. [4]. These principles aim to increase the ability of both humans and computers to understand data by making it Findable, Accessible, Interoperable, and Reusable (FAIR). These guiding principles have been the foundation for our automated process as we try to design a system to standardize the analysis of time series data across the whole solar field. In this paper, we propose a FAIRification framework for spatiotemporal data from PV power plants. We also propose automated methods for data quality assessment and missingness pattern classification that can be applied to time series PV data across the field. Investigating missingness patterns is essential for deciding imputation methods that can improve the model performance for degradation analysis studies. In this case, we examine missingness imputation through the application of several baseline methods including mean interpolation, linear interpolation, and KNNs.

II. Methods

A. PV Power Plants FAIRification

The data used in this project are stored in the SDLE research center's Apache Hadoop/Hbase/Spark cluster [5], which we will henceforth refer to as CRADLE (Common Research Analytics and Data Lifecycle Environment). This environment is based on the Cloudera CDH distribution. We use a Hadoop Distributed Filesystem (HDFS) to store all of our raw data. After cleaning, the data used for analysis is moved into the Apache Hbase. Hbase takes its inspiration from Google's Big Table, a NOSQL database based on triples where each observation in the dataframe has a rowkey and

This material is based upon work supported by the U.S. Department of Energy's Office of Energy Efficiency and Renewable Energy (EERE) under Solar Energy Technologies Office (SETO) Agreement Number DE-EE0009347 and DE-EE0009353. The views expressed herein do not necessarily represent the views of the U.S. Department of Energy or the United States Government.

a columnkey. In order to interact with CRADLE, we rely on Case Western's high performance computing cluster (HPC), an environment with over 250 compute servers, including more than 60 GPU nodes and 7000 processors. While the data used for this analysis is not made public yet, the data will be made accessible by the general PV community on OSF.io in accordance with FAIR guidelines.

We have developed a four-step data ingestion pipeline for receiving data from outside groups and ingesting it into the CRADLE ecosystem. Fig. 1 shows a visual representation of this process.

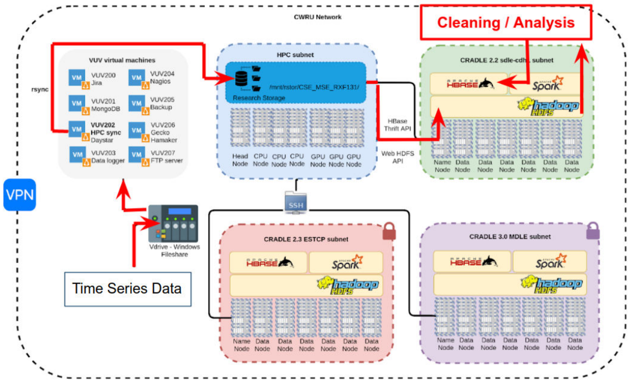


Fig. 1. Data Ingestion Pipeline

After receiving time series data, the first step is to move the data into the staging area. The staging area is stored in Case Western's V-drive, which is a Windows file sharing system hosted by the university. Every night, the contents of the V-drive get backed up automatically to the HPC for redundancy. Once the data is safe in the staging area, the next step is to move the raw data into Case's Hadoop Distributed File System (HDFS). At this point in the process, the raw data has been comfortably stored where it can be accessed should anyone ever need it again. The next step is to preprocess the data. This includes basic data cleaning, adding satellite weather data from SolarGIS, and metadata FAIRification. After the data has been processed, it is stored in an Apache Hbase table for ease of access for future analysis and modeling.

An especially important aspect of our data preprocessing step is metadata FAIRification. There are many benefits associated with FAIRifying our data. It makes our data more easily shareable with other groups because of the standards set for variable nomenclature and structure. It makes it easier for other groups to share data with us, as we can utilize our FAIRification framework to help computers understand more generally what certain variables mean. It also makes it easier to extract meaning from our modeling because of our structured, graph approach for our metadata. There has been an extensive push in the US to make metadata “FAIR” recently, as publishers, science funders, and government agencies have begun to establish requirements for the proper management of metadata. As such, we have been implementing FAIR principles into the ingestion of our data. Specifically, we use a standardized Javascript Object Notation for Linked Data (JSON-LD) filetype to store our metadata [6]. We have defined a new structure for our JSON-LD metadata files through the creation of a solar power plant ontology. In order to create and design our solar time series ontology, we have used the Protege ontology editor [7].

An ontology is a formal dictionary of terms for a given industry or field that shows how the terms are related through densely interconnected webs. Part of the point of doing this is to standardize terms for solar time series data by defining how variables should be defined across the industry. In our model, for example, latitude is to be spelled exactly latitude (not lat, latd, etc), and it is to be measured in degrees always. This way, there is no ambiguity. An ontology not only defines terms, but it defines a structure for the metadata as well. An ontology is the blueprint for linking metadata terms together through the creation of a knowledge graph. When an ontology is filled in with real data, it becomes a knowledge graph. An ontology is made through the creation of triples, or object-relationship pairs. Fig. 2 shows an example of a triple that connects a solar power plant to a latitude by the hasLatitude property. An ontology makes use of more general terms, defining how classes of objects relate to each other. This can allow a computer to understand generally what a variable means, which can help in its understanding of data received from other groups. A knowledge graph fills the classes from an ontology with values based on the structure defined by the ontology.

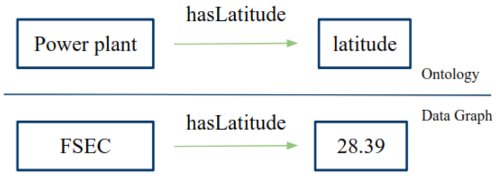


Fig. 2. Examples of Object-Relationship Pairs for the Ontology Blueprint and Resultant Graph

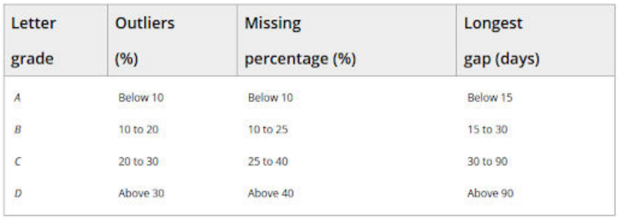
B. Data Quality Assessment

At SDLE, we have developed an R package for analyzing time series data called PVplr [8] that includes functions for the automated analysis of data quality. These functions include a heatmap generator and an automated data grading function. The data grading function assigns a letter grade for the data based on outliers, missingness percentage, and longest missing gap [3]. For a data set to receive an ”A” in all categories, it must have outliers less than 10%, missingness less than 10%, and a longest missing gap below 15 days, for example. A full outline of the metrics for the data quality grades can be seen in Fig. 3.

III. PV Power Plants Data sets Description

There are eight PV data sets that have been received from different companies and research institutions and have been ingested to the database of our research group. Table I lists

978-1-7281-6117-4/22/$31.00 ©2022 IEEE



Letter grade	Outliers (%)	Missing percentage (%)	Longest gap (days)
A	Below 10	Below 10	Below 15
B	10 to 20	10 to 25	15 to 30
C	20 to 30	25 to 40	30 to 90
D	Above 30	Above 40	Above 90

Fig. 3. Standards for PV time series grading system

some basic information about each data set. The PV systems in the same data set have the same meta variables and time series variables, but the PV systems in different data sets have some differences regarding both meta and time series variables. For example, meta information about the number of strings and the number of modules in each string exists in data set 1 but not in data set 2. The irradiance data in data set 1 is global horizontal irradiance, but it is plane of array irradiance in data set 2. The PV systems in different data sets can also refer to different scales. There are individual PV modules (such as in the data set 4), inverters for a PV array (such as in the data set 1), and inverters for a PV site (such as in the data set 2).

TABLE I
LIST OF PV SYSTEMS

ID	Average Age	# of Systems	Time Interval (minute)
1	8.24	354	15
2	1.42	1088	1
3	4.24	98	5
4	5.75	8	10
5	0.95	8	30
6	3.13	8	1
7	1.72	70	1
8	2.38	28	15

The data shared from system 8 is from the Florida Solar Energy Center and is the focus of this analysis. More specifically, data from PV systems from the SunSmart Schools program were made available. From inverters that control the racks of PV modules at these schools, we have many years of time series data logged. In total, 28 sites from this program have been shared from FSEC to the SDLE lab at Case Western. This data includes 15 minute interval time series data with information about power output, ambient and reference temperature, irradiance, battery properties, and input and output current and voltage. The length of the time series varies between schools, with the longest set including about 9 years of data, and most of the data sets including data on the order of about 2 years.

IV. Discussion

A. FAIRification

We have developed FAIRmaterials, both an R [9] and Python [10] package, for automating the creation of FAIRified JSON-LD files. Given a simple excel file of a user's metadata, these packages automatically generate a FAIRified JSON-LD file based off the standards that we have developed in our solar power plant ontology. We are currently going through all received data sets and collecting the variables provided with their typical names and units. This information will be provided to our collaborators, including both companies and research institutions, that own the time series PV data of multiple systems. Using feedback from these entities, we can improve our FAIRification process based off the input of real world users.

We have also developed an ontology to describe time series data for its application in solar. The steps we have taken for this process are outlined in Fig. 4.

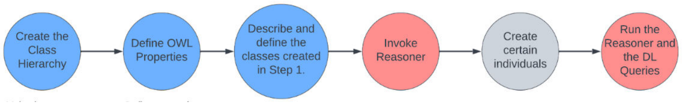

Create the Class Hierarchy
Define OWL Properties
Describe and define the classes created in Step 1.
Invoke Reasoner
Create certain individuals
Run the Reasoner and the DL Queries

Fig. 4. Steps in Ontology Design

The first step was to create a class hierarchy that describes all of the objects that need to exist in our ontology. In this case, we need to create a general class that describes a solar power plant. Each power plant will have information about its location, array, time series, and inverters. So we have chosen these to be the main sub-classes that describe our power plants. There is a visual depiction of this design in Fig. 5.

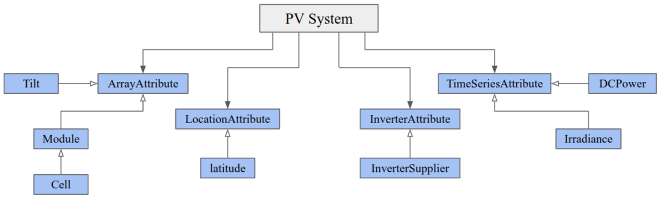


Fig. 5. Class Hierarchy for Solar Ontology

Given the structure from our class hierarchy, we can add more information to our graph by defining how our properties are connected. For example, properties can be described as functional if there is one unique value of y for each instance, x. So a PV power plant would only have one unique value for a longitude, making that a functional property. With a well defined hierarchy and correct property descriptions, we can make use of a reasoner in order to infer things about our data. As we begin to add instance level data into our ontology to create a knowledge graph, we can make use of the ontology's reasoning capabilities in order to discover important relationships in our data.

B. Data Grading

With the data from the SunSmart Schools program, we have performed an analysis of the data quality of a set of PV sites with the PVplr package [8]. In Fig. 6 we have generated a heatmap to visualize the quality of a representative data set.

We plot the time of day on the y-axis, with the date on the x-axis. The graph is then colored in by the power output. This kind of visualization is especially powerful for grading

978-1-7281-6117-4/22/$31.00 ©2022 IEEE

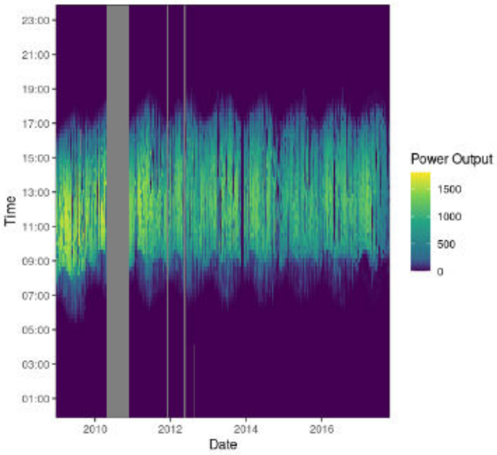


Fig. 6. Data Quality Heatmap

data quality, because missing chunks in the data are made especially apparent as grey bars. We can also see that the data that is not missing meets our expectations. We get high power output during the day when it is sunny, and no power output at night. Such analysis is important to perform at the beginning of a project in order to ensure that the data that we are working with is in line with reality. In Fig. 7 are heat maps plots of all the data that exist in our set. This application at scale is made easy through the use of the PVplr R package.

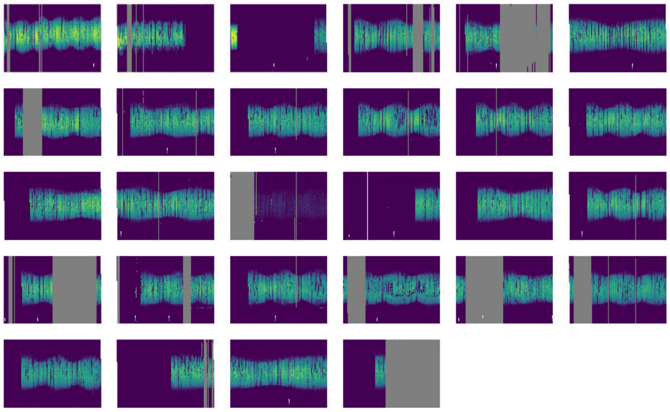


Fig. 7. Heatmaps for All of the Data in the Analysis Set

TABLE II
GRADED PV SYSTEMS

Site	Outlier Percentage	Missingness Percentage	Longest Missing Gap	Length Requirement
1	B	A	C	P
2	B	A	D	P
3	B	A	A	F
4	B	B	D	P
5	A	C	C	P
6	B	A	A	F
7	B	B	D	P
8	C	A	A	P
9	B	A	A	P
10	B	A	A	P
11	B	A	A	P
12	C	A	A	P
13	C	A	A	F
14	B	A	A	F
15	B	B	D	P
16	B	A	A	F
17	B	A	A	F
18	B	A	A	P
19	A	D	D	P
20	B	A	C	P
21	B	A	A	P
22	B	B	D	P
23	B	C	D	F
24	B	B	D	P
25	B	A	A	P
26	B	A	A	F
27	C	A	A	F
28	A	D	D	F

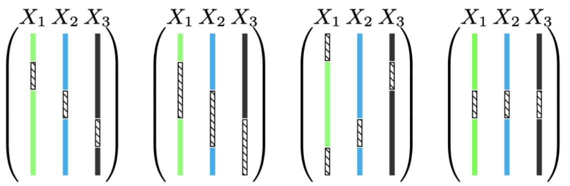
While a visual representation of the data is useful for human interpretation of the data quality, it's important to convert this into something that a computer can make sense of. We do this through assigning the data letter grades based on their percentage of outliers, missingness percentage, longest missing gap, and a pass/fail based on if the data is longer than two years or not. The grading function applied at scale on our data set is described in Table II. Using the results from our data grading process, we can easily decide which power plants provide more complete data for our analysis. This in turn allows us to focus in on the more important data sets that will allow for more in-depth analysis on our data. With an idea of the missingness existing in the data set, we can move towards trying to impute this missing data.

C. Missingness Pattern Discovery

We can characterize the patterns of missing values between the different PV time series data sets from two aspects. From a micro perspective, missing data in a series can be categorized into single or block. Single refers to a single missing value between known values while block refers to consecutive chunks of missing values. This is why our data grading function measures both missingness percent and longest missing gap. From a macro perspective between different power plants, depending on the positions of missing values, we consider four common missingness patterns: Missing Completely at Random (MCAR), Disjoint, Overlap, and Blockout, see Fig. 8 [11]. By identifying the common missingness patterns in our PV data sets, we can possibly construct suitable missing value imputation (MVI) models whose assumptions match our data sets.

D. Missing Data Imputation

In this section, we focus on 10 specfic sites in the Sunsmart schools data set that contain data at the same time over a one year period ranging from 09/01/2014 to 08/31/2015. We aim to impute the missingness of the module temperature and power readings from these sites. We measure imputation error



(a) Disjoint (b) Overlap (c) MCAR (d) Blackout

Fig. 8. Four Missingness Scenarios

by Mean Absolute Error (MAE) and Rooted Mean Squared Error (RMSE), defined as follows:

$$MAE = \frac{1}{m}\sum_{i=1}^{m}|P_i - \tilde{P}_i|; \quad RMSE = \sqrt{\frac{1}{m}\sum_{i=1}^{m}(P_i - \tilde{P}_i)^2} \tag{1}$$

where $m = card(M)$, $P_i \in P$, and $\tilde{P}_i \in \tilde{P}$, M is the set of missing data, P is set of imputed values, and $\tilde{P}$ is ground truth.

We compare the performance of three imputation methods. (1) *Linear Interpolation (LI)* [12]: a timeseries imputation method that fits a simple linear model using two values before and after the missing data block. Each missing data point will then be estimated using the linear model between these points. (2) *Mean Imputation (Mean)* [13]: a common approach that uses the column-wise mean to fill the missing data. (3) *K-nearest Neighbors (KNN)* [14]: imputes data by finding and averaging the K nearest neighbors to fill in the missing value.

To evaluate the accuracy of these three methods, we inject missing values into the real-world data sets. Particularly, we are interested in how different imputation methods perform when there are large chunks of missing values (Block Missing). To achieve this, we corrupt daily time series data of each PV system by randomly injecting a 16-hours block of missing values.

Our experiments have demonstrated the superiority of KNN over the LI and Mean methods for imputing missing values in the case of Block Missingness, see Fig. 9 and Fig. 10. KNN achieves a gain from 14.18% to 63.72% in imputation accuracy compared to LI and Mean. KNN likely outperforms other methods because it leverages spatial coherence from nearby systems while LI and Mean impute each PV system separately. The rich neighboring information within PV systems can improve the accuracy of imputation and potential predictive tasks like PV degradation rate prediction.

E. Spatiotemporal GNN Autoencoders

We propose the idea of using spatiotemporal GNN autoencoders to better leverage spatial coherence of PV systems and potentially further improve the imputation accuracy over KNN.

First, we need to translate PV systems into a graph. We map the PV systems into a spatiotemporal graph $G = (V, E, X_v(t))$ where nodes V represent PV systems, edges E are assigned using similarity or spatiotemporal correlations among PV systems, and $X_v(t)$ indicates node features. Since the locations of PV systems are fixed, the graph structure is static with time-invariant nodes and edges. However, $X_v(t)$ is time-varying. Each node consists of a set of time-series features such as power, irradiance and temperature and may have missing values in one or multiple of these features.

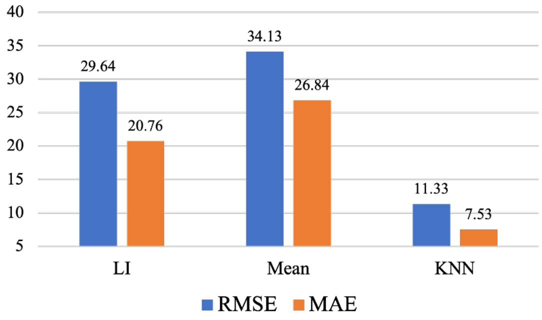


Fig. 9. Imputation Error for Module Temperature

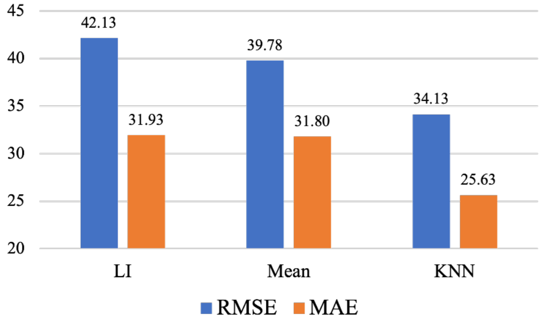


Fig. 10. Imputation Error for Power

Spatiotemporal Graph Neural Network modeling has demonstrated its performance improvement in power forecasting for PV power systems rather than utilizing an individual PV system [2] by capturing both spatial and temporal dependencies and coherence among PV systems. We propose a new framework - St-GNN Autoencoders (STGNN-AE) to detect and impute missing values in the PV data sets. Given a set of validated and correct data, we can learn a STGNN-AE, which consists of an encoder to transform inputs into a lower dimension representation and a decoder to recover the inputs from reduced data with a small reconstruction error. STGNN-AE will detect and localize erroneous values as outliers when observing a significant reconstruction error, suggest normal values to be used for imputation with the transformed values from reconstructed embedding by the decoder for local strategy, and provide synthetic PV input for simulation analysis over PV plants and regions when and where sensors are not available. The imputation quality will be measured by the impacts on the performance of downstream learning tasks like PV performance loss rate (PLR) prediction.

V. Conclusions

We have demonstrated in this paper the FAIRification of spatiotemporal PV time series data. By creating a solar power plant ontology, we propose standards for the naming and structure of metadata used to describe the data from these power plants. We can also use this ontology to assist in our modeling, where the computer can infer things about our data based on the relationships we have defined. Using the structure from this ontology, we have developed both R and Python packages for the automation of the FAIRification process. Going further, we have also developed an R package that automates the analysis of the quality of a data set through letter grades and heatmaps. We have shown that imputation methods that can leverage spatial coherence (e.g.: KNN) achieve higher imputation accuracy over simple methods like Linear and Mean Interpolation. To further improve the imputation accuracy, we propose the use of St-GNN autoencoders to detect and impute missing values from a data set by utilizing the spatial coherence between the power plants in the data set.

References

[1] J. Christiansen, "Global Market Outlook for Solar Power," SolarPower Europe, Tech. Rep., 2021.

[2] A. M. Karimi, Y. Wu, M. Koyuturk, and R. H. French, "Spatiotemporal graph neural network for performance prediction of photovoltaic power systems," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 35, no. 17, 2021, pp. 15 323–15 330.

[3] D. Moser, D. Bertani, A. J. Curran, R. H. French, M. Herz, and S. Lindig, "International Collaboration Framework for the Calculation of Performance Loss Rates: Data Quality, Benchmarks, and Trends," in *36th European Photovoltaic Solar Energy Conference and Exhibition*, Oct. 2019, pp. 1266–1271, citation Key Alias: moserInternationalCollaborationFramework2019a.

[4] M. D. Wilkinson, M. Dumontier, I. J. Aalbersberg, G. Appleton, M. Axton, and A. Baak, "The FAIR Guiding Principles for scientific data management and stewardship," *Scientific Data*, vol. 3, no. 1, pp. 1–9, Mar. 2016.

[5] Y. Hu, V. Y. Gunapati, P. Zhao, D. Gordon, N. R. Wheeler, M. A. Hossain, T. J. Peshek, L. S. Bruckman, G. Zhang, and R. H. French, "A Nonrelational Data Warehouse for the Analysis of Field and Laboratory Data From Multiple Heterogeneous Photovoltaic Test Sites," *IEEE Journal of Photovoltaics*, vol. 7, no. 1, pp. 230–236, Jan. 2017, tex.ids= hu2017nonrelational, huNonrelationalDataWarehouse2017a, huNonrelationalDataWarehouse2017b, huNonrelationalDataWarehouse2017c, hu_nonrelational_2017, huet.alNonrelationalDataWarehouse2017, yanghuNonrelationalDataWarehouse2017 conferenceName: IEEE Journal of Photovoltaics.

[6] A. Nihar, A. J. Curran, A. M. Karimi, J. L. Braid, L. S. Bruckman, M. Koyutürk, Y. Wu, and R. H. French, "Toward Findable, Accessible, Interoperable and Reusable (FAIR) Photovoltaic System Time Series Data," in *IEEE 48th PVSC Proceedings*, Jun. 2021.

[7] M. Horridge, R. S. Gonçalves, C. I. Nyulas, T. Tudorache, and M. A. Musen, "WebProtege: A Cloud-Based Ontology Editor," in *Companion Proceedings of The 2019 World Wide Web Conference*, ser. WWW '19. New York, NY, USA: Association for Computing Machinery, May 2019, pp. 686–689.

[8] A. Curran, T. Burleyson, S. Lindig, D. Moser, R. French, and SDLE Research Center, "PVplr: Performance Loss Rate Analysis Pipeline," Oct. 2020, tex.ids: a.j.curranPVplrSDLEPerformance2020, curranPVplrPerformanceLoss2020. [Online]. Available: https://CRAN.R-project.org/package=PVplr

[9] Willam C. Oltjen, Liangyi Huang, and Roger H. French, "FAIRmaterials: Make Materials Data FAIR," Sep. 2021. [Online]. Available: https://CRAN.R-project.org/package=FAIRmaterials

[10] Roger H. French, Liangyi Huang, William C. Oltjen, Arafath Nihar, Jiqi Liu, Justin Glynn, and Kehley Coleman, "Fairmaterials," Oct. 2021. [Online]. Available: https://pypi.org/project/fairmaterials/

[11] M. Khayati, A. Lerner, Z. Tymchenko, and P. Cudré-Mauroux, "Mind the gap: an experimental evaluation of imputation of missing values techniques in time series," in *Proceedings of the VLDB Endowment*, vol. 13, no. 5, 2020, pp. 768–782.

[12] T. Blu, P. Thévenaz, and M. Unser, "Linear interpolation revitalized," *IEEE Transactions on Image Processing*, vol. 13, no. 5, pp. 710–719, 2004.

[13] A. R. T. Donders, G. J. Van Der Heijden, T. Stijnen, and K. G. Moons, "A gentle introduction to imputation of missing values," *Journal of clinical epidemiology*, vol. 59, no. 10, pp. 1087–1091, 2006.

[14] R. Malarvizhi and A. S. Thanamani, "K-nearest neighbor in missing data imputation," *International Journal of Engineering Research and Development*, vol. 5, no. 1, pp. 5–7, 2012.

978-1-7281-6117-4/22/$31.00 ©2022 IEEE