SAND2022-XXXXR
LDRD PROJECT NUMBER: 227215
LDRD PROJECT TITLE: Deep learning-based spatio-temporal estimate of greenhouse gas emissions using satellite data
PROJECT TEAM MEMBERS: Hongkyu Yoon, Teeratorn Kadeethum, John Ringer, and Trevor Harris

ABSTRACT: Accurate estimation of greenhouse gases (GHGs) emissions is very important for developing mitigation strategies to climate change by controlling and reducing GHG emissions. This project aims to develop multiple deep learning approaches to estimate anthropogenic greenhouse gas emissions using multiple types of satellite data. NO_2 concentration is chosen as an example of GHGs to evaluate the proposed approach. Two sentinel satellites (sentinel-2 and sentinel-5P) provide multiscale observations of GHGs from 10-60m resolution (sentinel-2) to ~kilometer scale resolution (sentinel-5P). Among multiple deep learning (DL) architectures evaluated, two best DL models demonstrate that key features of spatio-temporal satellite data and additional information (e.g., observation times and/or coordinates of ground stations) can be extracted using convolutional neural networks and feed forward neural networks, respectively. In particular, irregular time series data from different NO_2 observation stations limit the flexibility of long short-term memory architecture, requiring zero-padding to fill in missing data. However, deep neural operator (DNO) architecture can stack time-series data as input, providing the flexibility of input structure without zero-padding. As a result, the DNO outperformed other deep learning architectures to account for time-varying features. Overall, temporal patterns with smooth seasonal variations were predicted very well, while frequent fluctuation patterns were not predicted well. In addition, uncertainty quantification using conformal inference method is performed to account for prediction ranges. Overall, this research will lead to a new groundwork for estimating greenhouse gas concentrations using multiple satellite data to enhance our capability of tracking the cause of climate change and developing mitigation strategies.

INTRODUCTION AND EXECUTIVE SUMMARY OF RESULTS:

Anthropogenic greenhouse gases (GHGs) emissions are the main cause of climate change. A recent Department of Homeland Security report (DHS, 2021) for addressing climate change emphasizes an increasingly urgent call to reduce GHG emissions to mitigate climate change. Hence, accurate estimation of GHG emissions is very important for developing mitigation strategies to climate change by controlling and reducing GHG emissions. However, the accurate estimation and measurement of spatio-temporal distributions of GHGs and other pollutants are difficult to achieve. Current conventional methods rely on point-based or surface-based measurements, which are typically limited to cover large areas and are temporally sporadic.

Over the past decade, a growing number of satellites with rapidly advancing technology provide numerous satellite data to fill in the information of GHGs in areas without ground-based

Sandia National Laboratories is a multimission laboratory managed and operated by National Technology and Engineering Solutions of Sandia, LLC, a wholly owned subsidiary of Honeywell International, Inc., for the U.S. Department of Energy's National Nuclear Security Administration under contract DE-NA-0003525.

monitoring data (e.g., Crisp et al., 2012; Crisp, 2015; Taylor et al., 2022). For examples, Orbiting Carbon Observatory (OCO-2/OCO-3, e.g., Taylor et al., 2020), ground-based data from Total Carbon Column Observing Network (TCCON), TROPOspheric Monitoring Instrument (TROPOMI) data [Finch et al., 2022], and multi-modal Sentinel satellite sensing data (e.g., Scheibenreif et al., 2021a) are available for enhancing remote sensing systems. Furthermore, multiple aspects of remote sensing data such as satellite and ground-based monitoring data need to be integrated to improve the estimation of spatio-temporal distribution of GHGs beyond a correlation approach.

Recent advances in deep learning approaches to analyzing satellite data show promising results to estimate greenhouse gas concentrations using remote sensing data (e.g., Yu and Liu, 2021; Scheibenreif et al., 2021a; Finch et al., 2022). For example, multi-band remote sensing data (Sentinel-2) and depth averaged NO_2 data (Sentinel-5P) have been used to estimate NO_2 concentration over Europe (Scheibenreif et al., 2021a). Especially this work is motivated by a deep learning approach in Scheibenreif et al. (2021a) where NO_2 data centered at ground-based monitoring station in Europe were processed from Sentinel-2 of multi-band spectral data (i.e., 12 channels) and Sentinel-5P of depth-averaged time-series data. However, two deep learning architectures with ResNet and convolutional neural network (CNN) in Scheibenreif et al. (2021a) are in parallel to extract features from two different data from sentinel-2 & sentinel-5P satellites, which are combined to estimate NO_2 distribution. However, multiscale/multimodal information is not thoroughly investigated to advance deep learning approaches to estimate anthropogenic GHG emissions using multiple types of satellite data.

In this work, we develop multiscale and multimodal machine learning algorithms to train models using two different satellite data (Sentinel-2 & Sentinel-5P) to estimate NO_2 concentrations. As in Scheibenreif et al. (2021a), average NO_2 concentrations over 2018-2020 and monthly data are used as training data and target output. For the average data, we first evaluate a CNN architecture than ResNet-50 (similar tunable parameters of ~25.6 M) in Scheibenreif et al. (2021a). Here we did not use a pretrained ResNet-50, but trained our model from random initialization. Our CNN architecture using the encoder part of U-Net (Ronneberger et al., 2015) performs slightly better than the ML architecture with pretrained weights used in Scheibenreif et al. (2021a), however, much better than that without pretrained weights in Scheibenreif et al. (2021a). This indicates that ML architecture (e.g., CNN) can be trained without relying on particular ML model architecture. However, given the small number of training data, a ML architecture with a smaller number of tunable parameters may perform better with proper hyperparameter optimization. In addition, coordinates of ground stations as additional input data to two Satellite data and CNN + long short-term memory (LSTM) for time-series data are evaluated and all of models we trained performed better than the original architecture in Scheibenreif et al. (2021a). Overall, this comparison study demonstrates that a certain class of neural network architectures can extract features of input data better, providing better performance. In addition, proper additional information such as coordinates of ground station or constraints improve model performance by

extracting additional features more efficiently. As well, changes in hyperparameters and model selection were found to improve performance.

For monthly data, we developed multiple deep neural network models. One of the best models is a deep neural operator architecture where channels of sentinel-2 (12 channels) and sentinel-5P (1 channel) are combined as input to a CNN architecture and time and coordinates of ground stations are another input to a feed forward neural network. A simple feature vector from each neural network block is concatenated to make a prediction of NO_2 concentration. Although the model architecture is simple, deep neural operator (DNO) architecture can stack time-series data as input, providing the flexibility of input structure without zero-padding. As a result, the DNO outperformed other deep learning architectures. We observe that temporal patterns featured with seasonal variation were predicted very well, while temporal features with frequent fluctuation were not predicted well. This observation indicates that a limited data point is one bottleneck to improve the prediction accuracy. In addition, uncertainty quantification using conformal inference method is performed to account for prediction ranges. Overall, this research will lead to a new groundwork for estimating greenhouse gas concentrations using remote sensing data including satellite data, which will enhance our capability of tracking the cause of climate change and developing mitigation strategies. This work will be submitted for journal publication (Teeratorn et al., in prep).

DETAILED DESCRIPTION OF RESEARCH AND DEVELOPMENT AND METHODOLOGY:

In this work we use the NO_2 concentration data processed in Scheibenreif et al. (2021a,b) where the sentinel-2 (S2) carrying a multi-spectral instrument provides satellite images of visible and shortwave-infrared wavelengths (a total of 12 channel data) at a fine scale (10-60m). The sentinel-5P (S5P) satellite carries differential optical adsorption spectroscopy to observe trace gases (e.g., NO_2, CH_4) in the atmosphere at a coarse scale (3-5km). In particular, the column density product of satellite data can be used to measure the temporal variation of surface NO_2 concentrations. These satellite data are used as input to the deep learning model which is optimized to match the ground-based measurement data of NO_2. For the period of 2018-2020 we investigate datasets which are averaged at two different temporal frequencies as described in the Addendum slide #1. For the average data S5P and NO_2 data at ground stations are averaged for the entire 2018-2020 timespan, whereas for “monthly” data S5P and NO_2 data are averaged over monthly intervals. Note that S2 images and other features (e.g., latitude/longitude) are considered static and remain constant for each station across different temporal frequencies. All data is normalized through z-score standardization (i.e., a mean of zero and a standard deviation of 1). In terms of data augmentation, during training S2 and S5P images are randomly rotated and flipped.

We present three representative models evaluated in this study and detailed description of model architectures is provided in the Addendum slide #1-#5. Briefly, Model 1 takes in both the Sentinel-2 (S2) image and the corresponding Sentinel-5P (S5P) data averaged over the entire observation period (2018-2020) as shown in the Addendum slide #2 and #5. S2 features are extracted with CNN block and S5P features are extracted using a simpler CNN architecture. The S5P network is smaller than the S2 network because the S5P data has a lower native resolution and only has a single channel (the S2 data has 12 channel data). It should be noted that the architecture of Model 1 with the encoder half of a U-Net model with dropout is based upon work by Scheibenreif et al. (2021a) where a Resnet-50 is used to extract features from S2 images. We also use a LeakyReLU activation function instead of a ReLU activation function where applicable. Both changes were found to modestly improve performance.

Model 2 with the time-gated LSTM (TGLSTM) model is used to predict time-series NO_2 concentration data (i.e., monthly data) (Addendum slide #3 & #5). The TGLSTM (Sahin and Kozat, 2018) is used instead of a regular LSTM layer because the TGLSTM can account for nonuniformly sampled timeseries data (i.e., data where the time between observations is not constant). This is done by incorporating time information as a nonlinear scaling factor using additional time gates. Here, "time information" is shown as Times in (slide #3), where Times is a tensor with 15 dimensions where each value is the time in months from October 2018 (the earliest recorded date for observations in the monthly dataset). For padded data a time of -1 is used. For timeseries additional data processing is used to account for the thousands of missing observations. Models expect inputs to be a uniform size, even though stations have a varying number of observations (see the Addendum slide #2). For Model 2 we decide to consider the first 15 observations for each station. If a station has less than 10 observations it is discarded, and for stations with 10 to 14 observations we pre-pad using zero-padding give the input a length of 15. Note that padded values are discarded from loss calculations and are not considered when updating model parameters.

Model 3 uses the deep neural operator concept (Lu et all, 2021) to construct model structure consisting of two networks; branch net of CNN and trunk net of feed-forward neural networks (FFN) (Addendum slide #4 & #6). The branch net takes all satellite data consisting of S2 12 channels and S5P single channel. Model 3 reduces these high dimensional inputs to reduced embedded manifolds of size 250. We note that the size of reduced embedded manifolds is one of the hyper-parameters that need to be explored during the training phase. However, to simplify our workflow, we fixed the dimension of reduced embedded manifolds to 250. The trunk net takes spatial and time coordinates of ground station NO_2 measurement and, again, collapses those data into the reduced embedded manifolds of size 250. Both reduced embedded manifolds are then combined through an element-wise addition and subsequently fed into a series of FNN to produce NO_2 prediction. For all three models, a loss function of the mean squared error (MSE) between measured and predicted NO_2 concentration is used.

RESULTS AND DISCUSSION:

We evaluate the performance of three deep learning models to predict NO_2 concentrations at air quality ground measurement station. For model evaluation, we use a randomly selected test dataset which is not used during training. The performance metrics include a goodness of fit (R^2) of linear regression, a mean absolute error (MAE), and a mean square error (MSE) (see the addendum slide #7). Results of three models in comparison with those in Scheibenreif et al. (2021a) are presented in the Addendum slide #8. For the average data case, Model 1 performance is very similar to that with a pretrained model and transfer learning in Scheibenreif et al. (2021a). Without the pretrained model , the model result in Scheibenreif et al. (2021a) is much worse (e.g., $R^2 = 0.38$) than our Model 1 ($R^2 = 0.55$). Note that we use only three model evaluation. Hence, the fact that the best model out of 3 runs performs similarly to that out of 10 runs in Scheibenreif et al. (2021a) demonstrates that our work tends to perform more robustly.

Because the GHG concentrations vary temporarily, it is important to evaluate model performance against time-series data. In this work, we evaluate our performance using the monthly dataset. A short description of the data is provided in the addendum slide #1 and the detailed description is available in Scheibenreif et al. (2021a,b). The Model 2 performs better than the model in Scheibenreif et al. (2021a) as shown in the addendum slide #8. However, the degree of the performance difference is relatively small. It should be noted that we did not use pre-trained weights nor transfer learning. However, ML models in Scheibenreif et al. (2021a) require pre-training and transfer learning to have a relatively good performance, which is still worse than our model 2 performance. On the other hand, the model 3 outperforms the model in Scheibenreif et al. (2021a) in all three measures. For an example, the R^2 and MAE values of the model 2 are 0.64 and 5.45 ($\mu g/m^3$), respectively, compared to 0.53 and 6.31 in Scheibenreif et al. (2021a). The better performance of Model 3 compared to Model 2 may stem from two different aspects of input and deep learning architecture. First, Model 3 uses the additional information of station coordinates (two-dimensional vector), while Model 2 does not. Second, the time-series data is stacked as an input to Model 3, hence Model 3 does not need to pad any missing monthly data with zero-padding, while Model 2 needs to make a uniform batch size of the time-series input due to the requirement in the LSTM. The first aspect can be easily tested by including additional information in Model 2. Our preliminary result shows that Model 2 with additional station coordinates improves the performance much better, close to the Model 3 performance in the testing set, while the training of Model 3 still performs much better (see the parity plot on the addendum slide #9).

The second aspect has been studied in the literature and the TGLSTM was developed to delineate the irregular time series data problem in the LSTM. However, in our case due to the limited dataset we still need to use a zero padding to the missing monthly data point. It is noted that we add the zero padding to the front of each dataset (i.e., using pre-padding), so the time-series information can be carried over to the end of the time considered properly through the LSTM unit. This last aspect highlights the architecture advantage of Model 3 where each time-

series data (i.e., satellite data) with a corresponding time can be used as input to two parallel deep learning blocks in Model 3 (see the addendum slide #4). We note that the size of feature vectors is one of the hyper-parameters that need to be explored during the training phase. However, to simplify our workflow, we use a constant size depending on the model.

We further analyze the model performance in terms of the characteristics of NO_2 concentration. After inspecting both good and bad prediction cases, we observe the notable difference between two cases as shown in the addendum slide #10. For many good prediction cases NO_2 concentration fluctuates more seasonally without local fluctuations. However, for bad prediction cases monthly fluctuations are notably observed. This analysis reveals that the limit in our prediction capability is predominantly governed by local scale patterns of data, which typically requires either big data or other constraints and physical description to improve machine learning based models. Overall, our models for the monthly data perform much better than the model in Scheibenreif et al. (2021a) and there is room to improve with hyperparameter optimization, larger dataset, and/or other physical constraints.

ANTICIPATED OUTCOMES AND IMPACTS:

This project aims to develop multiscale deep machine learning approaches to estimate anthropogenic greenhouse gas emissions using remote sensing data (e.g., satellite data). Accurate estimation of greenhouse gas emissions will enable us to develop proper mitigation strategies to decrease the anthropogenic sources contributing to climate change, which is a threat to global and national security. If successful, the proposed approaches can be readily applicable for filling gaps in the remote sensing data and rapid deployment for the upcoming new sensing data streams. This will impact the nation’s ability to advance the goals and objectives in the nation's Climate Action Plan.

Newly developed multiple deep learning models in this work improved prediction accuracy of spatio-temporal variations of NO_2 concentrations based on two different satellite images and additional information such as time of monthly data and coordinates of ground stations. Although we do not perform specific hyperparameter optimization, our analysis reveals several significant outcomes that can be readily applicable for remote sensing data. Due to data-preprocessing of Satellite data, we expect typical satellite data to have missing data points (e.g., cloud-based noise data). Some ML blocks such as LSTM requires the same input size in batch, requiring a zero-padding due to a limited dataset. Although it is trainable, the training efficiency to extract time-series information could be limited due to zero-padded datasets. On the other hand, the deep neural operator concept used in our Model 3 enables us to stack all time series data with corresponding times as model input, eliminating the zero-padding of missing data. This architectural advantage in Model 3 can be further developed as a tool for transfer learning in various remote sensing data. Due to a growing number of remote sensing data with multiscale and multimodal characteristics this portable architecture in the deep neural operator will improve

the integration of the multiple scale and multi-modal data very efficiently. If successful, this DL-based approach can be readily applicable for filling in gaps of remote sensing data and rapid deployment for upcoming new remote sensing data streams.

To further quantify the uncertainty of our trained model's predictions, we also use the recently developed split conformal inference method with our proposed positivity constraint. Conformal inference is a general framework for post-hoc uncertainty quantification for any, possibly black-box, prediction algorithm. Split conformal inference is a computationally efficient version that only requires a training and validation dataset, which are already standard practice in machine learning (see the addendum slide #11). The only issue with this approach is that the prediction interval is unbounded, whereas our targets are strictly positive. We modify the analysis procedure to produce strictly positive intervals using an invertible function (e.g., the inverse logit $g(x) = \log(\exp(x)-1)$ that continuously maps positive numbers to the real line. Due to its easiness and less restricted assumption, we can apply this approach for different datasets such as datasets featured with seasonal variations where we will be able to predict better and datasets featured with frequent fluctuations where we expect low accuracy of prediction. Hence, we can estimate the uncertainty bound more robustly depending on the model's performance.

This research will lead to a new groundwork for estimating greenhouse gas concentrations using remote sensing data including satellite data, which will enhance our capability of tracking the cause of climate change and developing mitigation strategies. This proposal aligns with DOE's science and energy goal of "Transformational research, development, demonstration and deployment of clean energy and efficiency technologies". The multiscale deep learning approaches also support many DOE research programs in which deep learning approaches can enhance nation's capabilities to improve our fast and reliable prediction in multimodal and multiscale problems that are common to many natural and engineered systems for climate, energy, and homeland security.

CONCLUSION:

Anthropogenic greenhouse gases (GHGs) emissions are the main cause of climate change. Accurate estimation of GHG emissions is very important for developing mitigation strategies to climate change by controlling and reducing GHG emissions. This project aims to develop multiple deep learning approaches to estimate anthropogenic greenhouse gas emissions using multiple types of satellite data. NO_2 concentration is chosen as an example of GHGs to evaluate the proposed approach. Two sentinel satellites (sentinel-2 and sentinel-5P) provide multiscale observations of GHGs from 10-60m resolution (sentinel-2) to ~kilometer scale resolution (sentinel-5P). In particular, the column density product of satellite data can be used to measure the temporal variation of surface NO_2 concentrations.

Multiscale aspects of satellite data are used as input to multiple architectures of a combination of convolutional neural networks and feed forward neural network. Additional information such as

the coordination of ground stations and time for time-series data are used as input to feed forward neural network. Extracted features from multiple input are concatenated to predict NO_2 concentrations through output layers or LSTM architecture. Overall, we demonstrate that multiple models outperform those in the literature and especially our model performance is much better with more challenging monthly dataset in which a long-term variations such as seasonal changes and short-term fluctuation may cause a different level of challenges for model development. Due to the rapid execution time of trained deep learning model compared to any existing models, this deep learning approach can be combined with increasing remote sensing campaign and high-fidelity model such as evaluation of atmospheric carbon dioxide concentrations as simulated by the Energy Exascale Earth System Model (E3SM). Overall, this research will lead to a new groundwork for estimating greenhouse gas concentrations using remote sensing data including satellite data, which will enhance our capability of tracking the cause of climate change and developing mitigation strategies.

REFERENCES:

Crisp, D., 2015, Measuring atmospheric carbon dioxide from space with the Orbiting Carbon Observatory-2 (OCO-2), SPIE Optical Engineering + Applications, San Diego, California, United States, 8 September 2015, 960702, https://doi.org/10.1117/12.2187291.

Crisp, D., et al., 2012, The ACOS CO2 retrieval algorithm – Part II: Global XCO2 data characterization, Atmos. Meas. Tech., 5, 687–707, https://doi.org/10.5194/amt-5-687-2012.

DHS, 2021, DHS Strategic Framework for Addressing Climate Change, https://www.dhs.gov/sites/default/files/publications/dhs_strategic_framework_10.20.21_final_508.pdf.

Taylor, T.E., et al., 2022, An 11-year record of XCO$_2$ estimates derived from GOSAT measurements using the NASA ACOS version 9 retrieval algorithm, Earth Syst. Sci. Data, 14, 325–360, https://doi.org/10.5194/essd-14-325-2022.

Finch, D.P., Palmer, P.I. and Zhang, T., 2022. Automated detection of atmospheric NO2 plumes from satellite data: a tool to help infer anthropogenic combustion emissions. Atmospheric Measurement Techniques, 15(3), pp.721-733.

Kadeethum, T., Ringer, R.J., T. Harris, and Yoon, H., Multiscale Deep Learning Approaches for estimation of greenhouse gas concentrations using multiple satellite sensing data (in prep).

Lu, L., Jin, P., Pang, G., Zhang, Z., & Karniadakis, G. E. (2021). Learning nonlinear operators via DeepONet based on the universal approximation theorem of operators. Nature Machine Intelligence, 3(3), 218-229.

Ronneberger, O., Fischer, P. and Brox, T., 2015, October. U-net: Convolutional networks for biomedical image segmentation. In International Conference on Medical image computing and computer-assisted intervention (pp. 234-241). Springer, Cham.

Sahin, S.O. and Kozat, S.S., 2018. Nonuniformly sampled data processing using LSTM networks. IEEE transactions on neural networks and learning systems, 30(5), pp.1452-1461. Pytorch implementation at https://github.com/FedericOldani/TGLSTM.

Scheibenreif, Linus M., Mommert, Michael and Borth, Damian, 2021a, Estimation of Air Pollution with Remote Sensing Data: Revealing Greenhouse Gas Emissions from Space, ICML 2021 Workshop on Tackling Climate Change with Machine Learning, https://www.climatechange.ai/papers/icml2021/23. code available at https://github.com/HSG-AIML/RemoteSensingNO2Estimation.

Scheibenreif, Linus, Mommert, Michael, & Borth, Damian, 2021b. Estimation of Air Pollution with Remote Sensing Data: Revealing Greenhouse Gas Emissions from Space [Data set]. Tackling Climate Change with Machine Learning Workshop at ICML 2021. Zenodo. https://doi.org/10.5281/zenodo.5764262

Taylor et al., 2020, OCO-3 early mission operations and initial (vEarly) XCO2 and SIF retrievals, Remote Sensing of Environment, Volume 251, 112032, https://doi.org/10.1016/j.rse.2020.112032.

Yu, M. and Liu, Q., 2021. Deep learning-based downscaling of tropospheric nitrogen dioxide using ground-level and satellite observations. Science of The Total Environment, 773, p.145145.

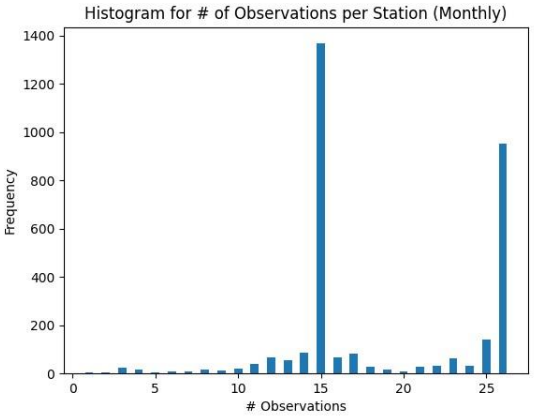
LABORATORY DIRECTED RESEARCH & DEVELOPMENT

WHERE INNOVATION BEGINS

ADDENDUM:

1 TIMESERIES DATA

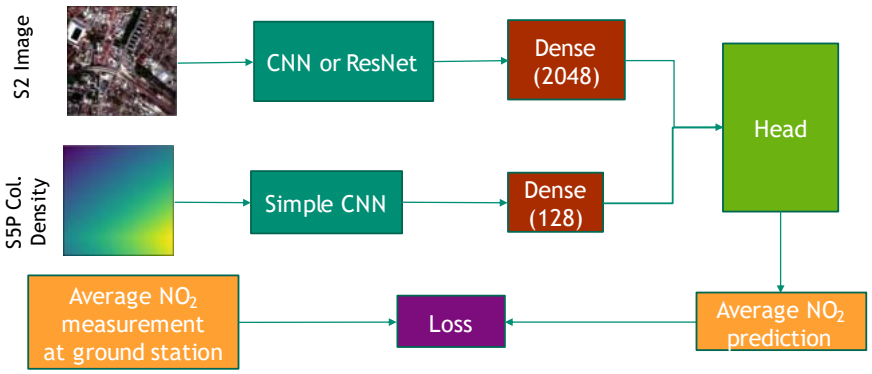
- Overall monthly dataset contains 59,563 data points (NO_2 recordings/S5P image) from 3,185 stations (Scheibenreifet al., 2021b).
  - Each station has a single S2 image, which is considered static.
- However, many of these stations don't have complete information from the 2018-2020 timespan (see plot on the right).
- 105 of these stations have less than 10 recordings and these are excluded when training/evaluating models (Models 1 & 2).
- Model 2 require timeseries to be of the same length for LSTM Model 2
  - Zero-pad stations with <15 observations
  - Use first 15 recordings for stations with >15
  - We plan to modify this approach in the future
- All time series data are stacked for Model 3, so Model 3 does not have to do zero-padding



All data used in this work is downloaded from Scheibenreifet al. (2021b), https://zenodo.org/record/5764262#.YfiPS1XYUs. The original work is in Scheibenreifet al. (2021a).

2 MACHINE LEARNING ARCHITECTURE (MODEL 1)

- Model consists of 2 CNN architectures (+ dense layer) followed by a "head"
- Head consists of a dense layer followed by an activation function, followed by a final dense layer for generating a single NO_2 prediction for the location.
- Feature vectors of dense layer size are hyperparameters

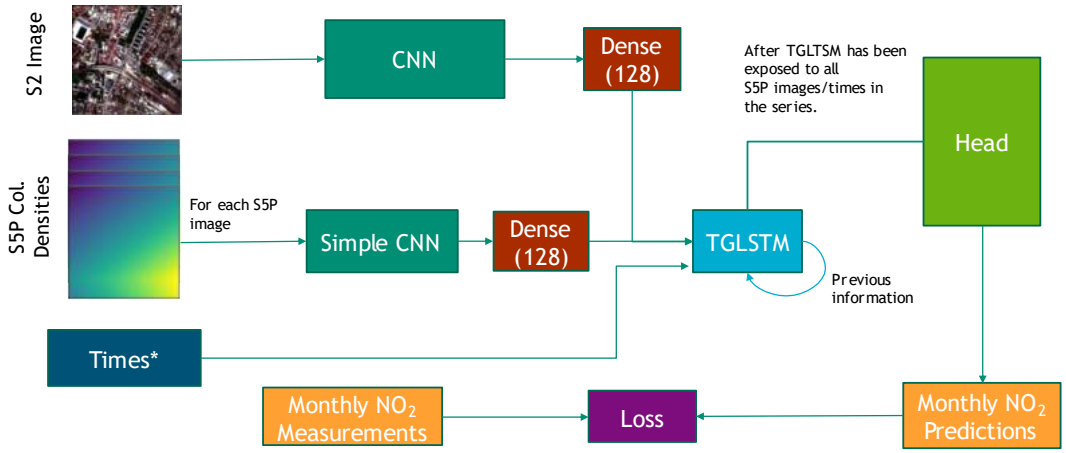


LABORATORY DIRECTED RESEARCH & DEVELOPMENT

WHERE INNOVATION BEGINS

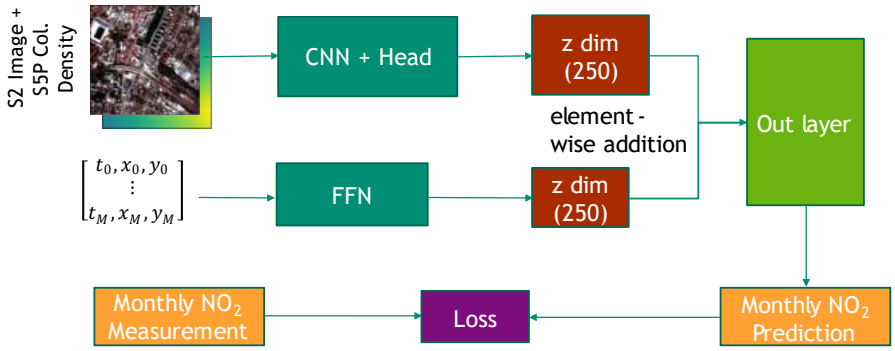
3 MACHINE LEARNING ARCHITECTURE (MODEL 2)

- Time-Gated Long Short Term Memory (TGLSTM, Sahin and Kozat, 2018)
- Sentinel 2, S5P column density, and time data are used as separate input
- Time-Gated LSTM recursively processes time-series data



4 MACHINE LEARNING ARCHITECTURE (MODEL 3)

- Model consists of 1 CNN followed by a "head" and 1 Feedforward Neural Network (FNN) architectures
- Head consists of a dense layer followed by an activation function
- Both branches are reduced to z reduced dimensions before we perform element-wise addition; then out layer delivers NO_2 prediction



LABORATORY DIRECTED RESEARCH & DEVELOPMENT

WHERE INNOVATION BEGINS

5 MODEL 1&2 ARCHITECTURE COMPONENTS

CNN with S2 image

block	input size	output size	instance normalization	dropout
1st contracting block	[B, 12, 128, 128]	[B, 128, 128, 128]	x	
2nd contracting block	[B, 12, 128, 128]	[B, 256, 64, 64]	x	
3rd contracting block	[B, 256, 64, 64]	[B, 512, 32, 32]	x	x
4th contracting block	[B, 512, 32, 32]	[B, 512, 16, 16]	x	x
5th contracting block	[B, 512, 16, 16]	[B, 1024, 8, 8]	x	x
AdaptiveAvgPool2d	[B, 1024, 8, 8]	[B, 1024, 1, 1]		
Flatten	[B, 1024, 1, 1]	[B, 1024]		

*B represents batch size. Each contracting block consists of 2 conv blocks, where each conv block is a Conv2d layer (kernel of 3, stride of 2) followed by an instance normalization layer, which are then followed by MaxPool layer.

Simple CNN with S5P image

block	input size	output size	instance normalization	dropout
Conv2d-1	[B, 1, 120, 120]	[B, 10, 118, 118]		
LeakyReLU	[B, 10, 118, 118]	[B, 10, 118, 118]		
MaxPool	[B, 10, 118, 118]	[B, 10, 39, 39]		
Conv2d-2	[B, 10, 39, 39]	[B, 15, 35, 35]		
MaxPool	[B, 15, 35, 35]	[B, 15, 11, 11]		
Flatten	[B, 15, 11, 11]	[B, 1815]		

Head

block	input size	output size	instance normalization	dropout
Linear	[B, 2176]	[B, 544]		
LeakyReLU	[B, 544]	[B, 544]		
Linear	[B, 544]	[B, 1]		

TGLSTM

block	input size	output size	instance normalization	dropout
LSTMCell*	[B, 256], [B, 1], ([B, 256], [B, 256])	[B, 256]		

*Note that the LSTMCell block is called recursively for each timestep in the data (in our case 15 times). This is why the input sizes are [B, features] instead of [B, 15, n_features].

6 MODEL 3 ARCHITECTURE COMPONENTS

CNN with S2/S5P images

block	input size	output size	instance normalization	dropout
1st convolutional layer	[B, 1, 128, 128]	[B, 32, 128, 128]		
1st contracting block	[B, 32, 128, 128]	[B, 64, 64, 64]	x	x
2nd contracting block	[B, 64, 64, 64]	[B, 128, 32, 32]	x	x
3rd contracting block	[B, 128, 32, 32]	[B, 256, 16, 16]	x	x
4th contracting block	[B, 256, 16, 16]	[B, 512, 8, 8]	x	
5th contracting block	[B, 512, 8, 8]	[B, 1024, 4, 4]	x	
6th contracting block	[B, 1024, 4, 4]	[B, 2048, 2, 2]	x	
7th contracting block	[B, 2048, 2, 2]	[B, 4196, 1, 1]	x	
1st bottleneck	reshape([B, 4196])	[B, 512]		

*B represents batch size. A contracting block that performs two convolutions followed by a max pool operation. The bottleneck employs two linear layers.

HEAD

block	input size	output size	instance normalization	dropout
1st linear layer	[B, 512]	[B, 256]		x
2nd linear layer	[B, 256]	[B, 250]		

*The first linear layer is subjected to LeakyRelu, but the second one is not subjected to any activation functions.

FFN

Block	input size	output size	instance normalization	dropout
1st linear layer	[B, 3]	[B, 256]		
2nd linear layer	[B, 256]	[B, 256]		
3rd linear layer	[B, 256]	[B, 256]		
4th linear layer	[B, 256]	[B, 256]		
5th linear layer	[B, 256]	[B, 256]		
6th linear layer	[B, 256]	[B, 250]		

Out layer

Block	input size	output size	instance normalization	dropout
1st linear layer	[B, 256]	[B, 1]		

7 RESULT METRICS

- r^2 –coefficient of determination between NO_2 predictions and actual ground measurements.
- MAE – Mean absolute error.
- MSE – Mean squared error.
- X-Mean – average of scores for X metric over a number of trained model predictions.
- X-T# – Best score for X metric over a number (#) of trained model predictions

$$r^2 = 1 - \frac{RSS}{TSS} = \frac{\sum_{i=1}^{n}(y_i - \hat{y}_i)^2}{\sum_{i=1}^{n}(y_i - \bar{y})^2}$$

n - # of NO_2 observations
y_i - NO_2 measurement for station/time i
$\hat{y}_i$ - NO_2 prediction for station/time i
$\bar{y}$ - Average of all NO_2 measurements

$$MAE = \frac{1}{n}\sum_{i=1}^{n}|y_i - \hat{y}_i| \qquad MSE = \frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2$$

8 TECHNICAL RESULTS

Original results from Scheibenreif et al. (2021a)*

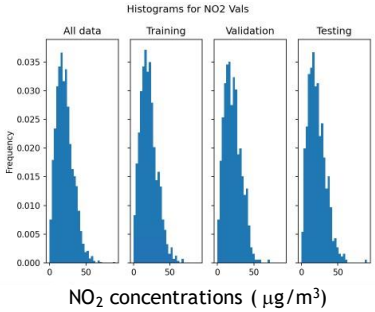
DATA	TIME	N-OBS.	PT	R2	R2-T10	MAE	MAE-T10	MSE	MSE-T10
SEN.-2,5P	2018-20	3.1K	✓	**0.54±0.04**	**0.59**	**5.92±0.44**	**5.42**	**62.52±5.47**	**56.28**
SEN.-2,5P	QUART.	19.6K	✓	0.52±0.05	0.57	6.24±0.22	5.98	73.1±6.88	66.12
SEN.-2,5P	MONTH.	59.6K	✓	0.51±0.01	0.53	6.54±0.15	6.31	78.96±4.2	73.74

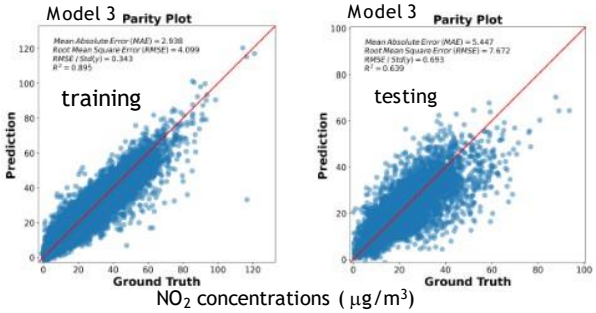
*Two cases in red box are compared with deep learning models. In Scheibenreifet al. (2021a), a total of 10 different models with random initial weights are trained. T10 is the best top model out of 10 models.

Model#	Ensemble*?	N-Obs	R2 (Mean)	R2-T3	MAE	MAE-T3	MSE (Mean)	MSE-T3
Model 1	Y	3.1k	0.55	0.61	5.84	5.43	59.33	56.3
Model 2 (CNN+TGLSTM)	Y	59.6k	0.56	0.58	6.43	6.23	79.9	71.9
Model 3 (CNN+FNN, station (t,x,y))	N	59.6k	0.64		5.45		58.86	

All models used for evaluation against test dataset are selected based on the best model in the validation loss during the training phase.
* For multiple runs, we perform only three models trained. T3 is the best top model out of 10 models. No ensemble represents single trained model.

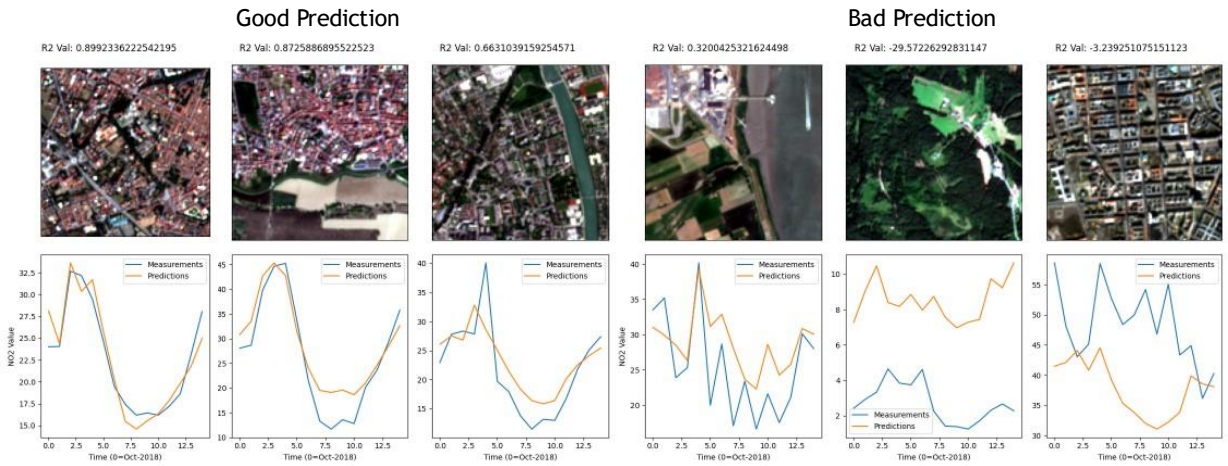
9 TECHNICAL RESULTS: TRAINING & TESTING





- Data distribution in training, validation, and testing datasets are similar (left plot)
- Parity plots for training and testing data from Model 3 are shown with key performance metrics (right plots).

10 TECHNICAL RESULTS: PREDICTION CASES (MODEL 2)

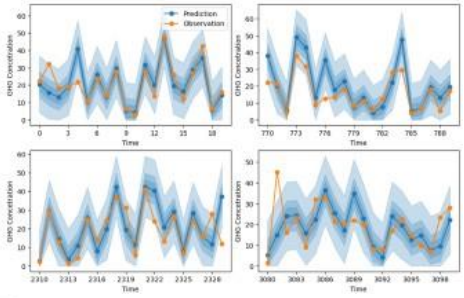


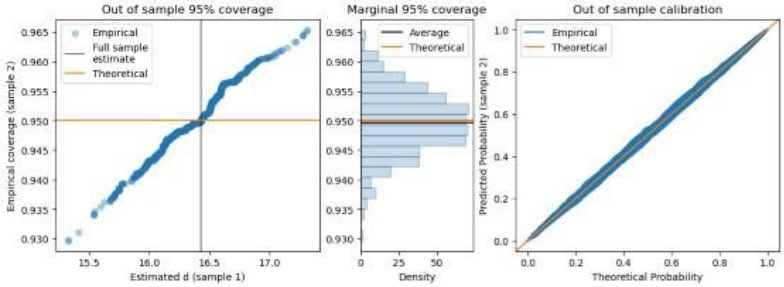
- With monthly data, NO_2 data with smooth changes is predicted better than those with more frequent changes
- Smooth pattern in changes may reflect seasonal variations (winter is high and summer is low)

11 TECHNICAL RESULTS: UNCERTAINTY QUANTIFICATION

Quantify uncertainty with **conformal inference**

- Generates well-calibrated prediction intervals for any prediction algorithm with Model 3
- Only requires a training and test dataset
- Compute test residuals $r_i = |y_i - \hat{y}_i|$ for $i \in 1, \ldots, N_{test}$
- Prediction interval $\hat{y} \pm d_\alpha$ where $d_\alpha = Quantile(1 - \alpha, r_1, \ldots, r_{N_{test}})$





Left – Estimated $d_{0.05}$ against out of sample coverage. Concentrates around the target 95%.

Right – Out of sample calibration. Blue lines are estimates. Orange line is perfect calibration.

12 REFERENCES

Scheibenreif, Linus M., Mommert , Michael and Borth, Damian, 2021a, Estimation of Air Pollution with Remote Sensing Data: Revealing Greenhouse Gas Emissions from Space, ICML 2021 Workshop on Tackling Climate Change with Machine Learning, https://www.climatechange.ai/papers/icml2021/23. code available at https://github.com/HSG - AIML/RemoteSensingNO2Estimation.

Scheibenreif, Linus, Mommert , Michael, & Borth, Damian, 2021b. Estimation of Air Pollution with Remote Sensing Data: Revealing Greenhouse Gas Emissions from Spac[Data set]. Tackling Climate Change with Machine Learning Workshop at ICML 2021. Zenodo. https://doi.org/10.5281/zenodo. 5764262.

Sahin, S.O. and Kozat, S.S., 2018. Nonuniformly sampled data processing using LSTM networks. IEEE transactions on neural networks and learning systems, 30(5), pp.1452 -1461. Pytorch implementation available at https://github.com/FedericOldani/TGLSTM.

Lu, L., Jin, P., Pang, G., Zhang, Z., & Karniadakis, G. E. (2021). Learning nonlinear operators via DeepONet based on the universal approximation theorem of operators Nature Machine Intelligence, 3(3), 218 -229.

Kadeethum, T., Ringer, R.J., T. Harris, and Yoon, H., Multiscale Deep Learning Approaches for estimation of greenhouse gas concentrations using multiple satellite sensing data(in prep).