Reconstruction of Total Solar Irradiance by Deep Learning

Table of Contents

Author(s)

- Author1 = {"name": "Yasser Abduallah", "affiliation": "Department of Computer Science, New Jersey Institute of Technology", "email": "ya54@njit.edu (mailto:ya54@njit.edu)", "orcid": "https://orcid.org/0000-0003-0792-2270"} (https://orcid.org/0000-0003-0792-2270"%7D)
- Author2 = {"name": "Jason T. L. Wang", "affiliation": "Department of Computer Science, New Jersey Institute of Technology", "email": "wangj@njit.edu (mailto:wangj@njit.edu)", "orcid": "https://orcid.org/0000-0002-2486-1097"} (https://orcid.org/0000-0002-2486-1097"%7D)
- Author3 = {"name": "Yucong Shen", "affiliation": "Department of Computer Science, New Jersey Institute of Technology", "email": "ys496@njit.edu (mailto:ys496@njit.edu)", "orcid": ""}
- Author4 = {"name": "Khalid A. Alobaid", "affiliation": "Department of Computer Science, New Jersey Institute of Technology", "email": "kaa65@njit.edu (mailto:kaa65@njit.edu)", "orcid": ""}
- Author5 = {"name": "Serena Criscuoli", "affiliation": "National Solar Observatory, Boulder, Colorado, USA", "email": "scriscuo@nso.edu (mailto:scriscuo@nso.edu)", "orcid": "0000-0002-4525-9038"}

- Author6 = {"name": "Haimin Wang", "affiliation": "Institute for Space Weather Sciences, New Jersey Institute of Technology", "email": "haimin.wang@njit.edu (mailto:haimin.wang@njit.edu)", "orcid": "https://orcid.org/0000-0002-5233-565X"} (https://orcid.org/0000-0002-5233-565X"%7D)

Purpose

The Earth's primary source of energy is the radiant energygenerated by the Sun, which is referred to as solar irradiance,or total solar irradiance (TSI) when all of the radiation is mea-sured. A minor change in the solar irradiance can have a sig-nificant impact on the Earth's climate and atmosphere. As aresult, studying and measuring solar irradiance is crucial inunderstanding climate changes and solar variability. Severalmethods have been developed to reconstruct total solar irra-diance for long and short periods of time; however, they arephysics-based and rely on the availability of data, which doesnot go beyond 9,000 years. In this paper we propose a newmethod, called TSInet, to reconstruct total solar irradiance bydeep learning for short and long periods of time that span be-yond the physical models' data availability. On the data thatare available, our method agrees well with the state-of-the-artphysics-based reconstruction models. To our knowledge, thisis the first time that deep learning has been used to reconstructtotal solar irradiance for more than 9,000 years.
To our knowledge, this is the first time that TSI reconsturction is made deep learning.

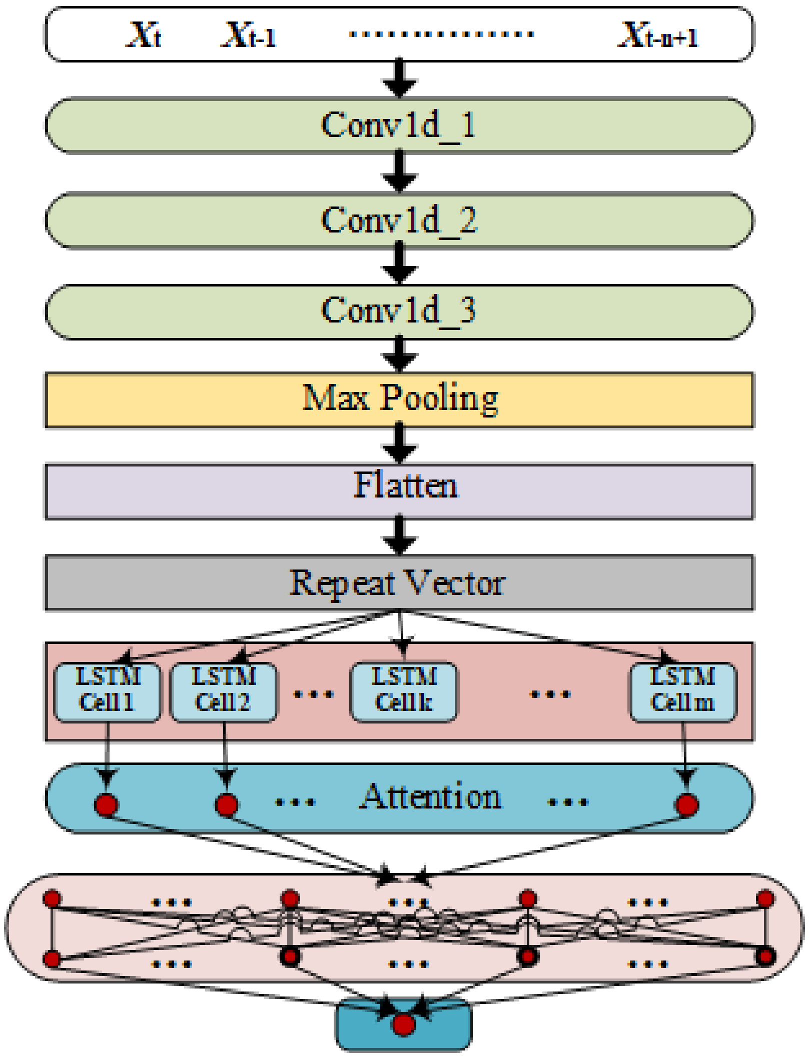
In this notebook we provide an overview of the TSInet system to demonstrate how to reconstruct the TSI using deep learning (DL) that goes beyond the phsyical parameters construction limit.

Technical Contributions

- We provide the community with a new tool to reconstruct the total solar irradiance (TSI) for 9,000 years and byone.

Methodology

The figure below presents the architecture of our TSInet model. TSInet is created using the tensorflow keras framework. Let *t* be the latest time point. Data sample x_t contains *w* values $v_t, v_{t-1}, \ldots, v_{t-w+1}$ and the label v_{t-w} where v_t is the TSI value at time point *t*. In our study, the time window, *w*, is set to 7. We train TSInet with multiple batches. In the first batch, we use the *n* training data samples, $x_t, x_{t-1}, \ldots, x_{t-n+1}$, to train TSInet. In our study, the number of input data samples,*n*, is set to 10. The label of the *n* training data samples is determined by the label of the last data sample (i.e.,x_{t-n+1}).In the second batch, we use the next n training data samples, $x_{t-n}, x_{t-n-1}, \ldots, x_{t-2n+1}$, to train TSInet. The label of the n training data samples is determined by the label of x_{t-2n+1}. In the third batch, we use the following *n* training data samples,*$x_{t-2n}, x_{t-2n-1}, \ldots, x_{t-3n+1}$, to train TSInet. The label of the *n* training data samples is determined by the label of xt−3n+1. We continue this training process until all TSI values in the training set are used. Forevery two adjacent data samples x_i, x_{i-1}, they overlap on $w-1$ TSI values, namely $v_{i-1}, v_{i-2}, \ldots, v_{i-w+1}$. TSInet consists of three convolutional layers (Conv1d_1,Conv1d_2, and Conv1d_3) where the kernel slides along 1-dimension on the time series, a max pooling layer, a flatten layer, a repeat vector layer, an LSTM (long short-termmemory) layer, an attention layer, two fully connected layers, and an output layer. The output from the three convolutional layers is flattened by the flatten layer and transformed into a sequence, also known as a feature vector. The repeatvector layer repeats the feature vector to reshape and prepare it as the input to the LSTM layer. The LSTM layer in our architecture contains *m* LSTM cells (in this study,*m* is set to 10). The attention layer with *m* neurons is used to focus on the relevant information in each time step. Each of the two fully connected layers has 200 neurons. The activation function used in our model is ReLU (rectified linear unit). TSInet produces as output a predicted TSI value.



TSInet Contextual Architecture.

Funding

N/A

Keywords

keywords=["TSI","solar","irradiance","Reconstruction", "Prediction", "Machine", "Learning"]

Citation

To cite this notebook: Yasser Abduallah, Jason T. L. Wang, Yucong Shen, Khalid A. Alobaid, Serena Criscuoli, Haimin Wang. Reconstruction of Total Solar Irradiance by Deep Learning available at: https://github.com/ccsc-tools/TSInet-irradiance-

reconstruction/YA_01_ReconstructionOfTSIbyDeepLearning.ipynb (https://github.com/ccsc-tools/TSInet-irradiance-reconstruction/blob/main/YA_01_ReconstructionOfTSIbyDeepLearning.ipynb).

Acknowledgements

We acknowledge the use of TSI provided by the Total Irradiance Monitor aboard the SOlar Radiation and Climate Experiment (SORCE) and available at: <a target='_new' hred='http://lasp.colorado.edu/home/sorce/data/tsi-data>http://lasp.colorado.edu/home/sorce/data/tsi-data (http://lasp.colorado.edu/home/sorce/data/tsi-data>http://lasp.colorado.edu/home/sorce/data/tsi-data).

Setup

Installation on Local Machine

Running this notebook in a local machine requires Python version 3.6.x or 3.8.x with the following packages and their version:

Library	Version	Description
keras	2.3.0	Deep learning API
keras_self_attention	latest	Self attention mechanism
numpy	1.21.5	Array manipulation
sklearn	latest	Tools for predictive data analysis
pandas	1.3.4	Data loading and manipulation
scipy	1.7.1	Provides algorithms for optimization and statistics
tensorflow	1.14 or 2.4	Provides the visualization and tooling needed for machine learning
tensorflow-gpu	1.14 or 2.4	Deep learning tool for high performance computation

Library Import

The following libraries need to be imported.

In [1]:
```
#supress warning messages
import warnings
warnings.filterwarnings('ignore')
print('Importing packages..')
# Data manipulation
import pandas as pd
import numpy as np
import os
print('Packages imported')
```

```
Importing packages..
Packages imported
```

Data Processing and Analysis ¶

We use measurements of the TSI provided by the Total Irradiance Monitor aboard the SOlar Radiation and Climate Experiment (SORCE)(Rottman2005) and available at: http://lasp.colorado.edu/home/sorce/data/tsi-data/ (http://lasp.colorado.edu/home/sorce/data/tsi-data/). This dataset, used as our training set, contains daily TSI measurements carried outfrom 2003 to the present (2021). Figure 1 illustrates the SORCE time series dataset showing the total solar irradiance overtime.

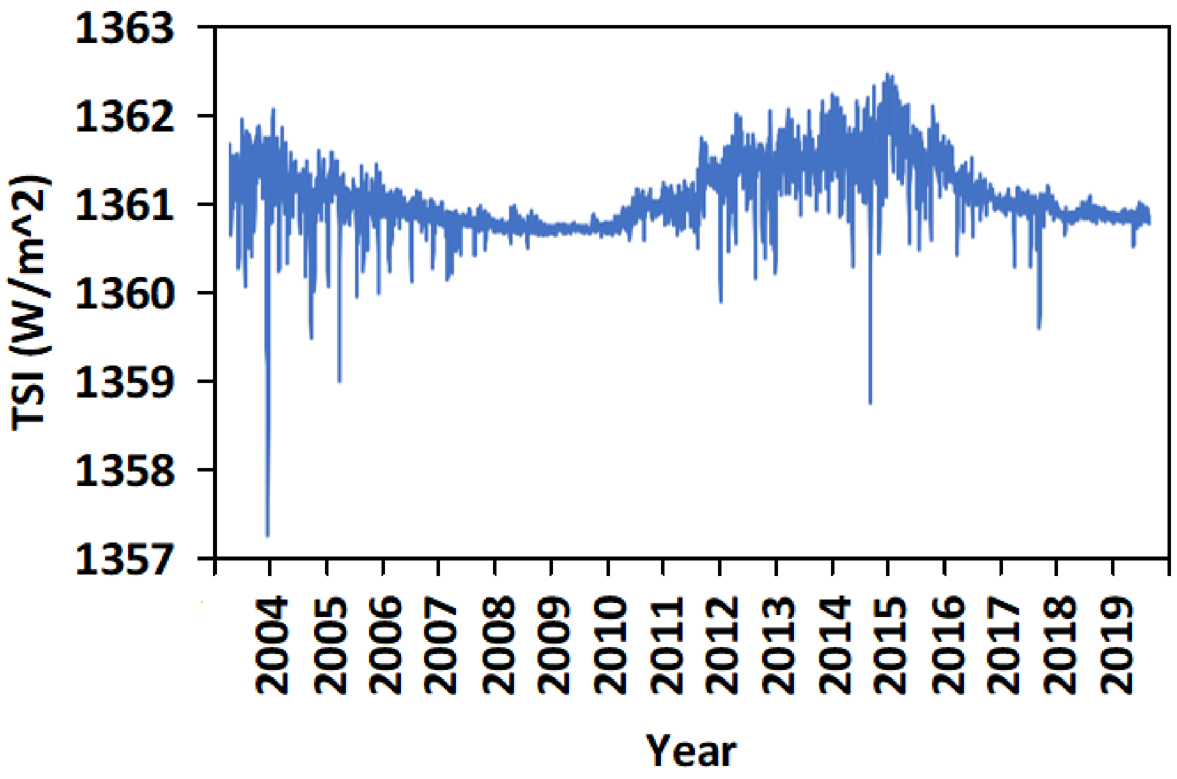


Figure 1: SORCE total solar irradiance (TSI) data from 2003 to the present (2021).

Our testing set contains measurements from TCTE TotalSolar Irradiance daily averages, available athttp://lasp.colorado.edu/lisird/data/tcte_tsi_24hr/. Total Solar Irradiance Calibration Transfer Experiment (TCTE) (http://lasp.colorado.edu/home/missions-projects/quick-facts-tcte/) mea-surements are made by the LASP TCTE Total Irradiance Monitor (TIM) instrument aboard the U.S. Air Force's STPSat-3 spacecraft. This TIM has been measuring total solar irradiance since late 2013. In addition, we adopt the following publicly available datasets obtained, over different temporal ranges, by different physics-based models, which will are used as testing setsin our work:

- NRLTSI2 (Naval Research Lab-oratory's TSI-2) Daily Averages, available at: http://lasp.colorado.edu/lisird/ (http://lasp.colorado.edu/lisird/)
- SATIRE-S (Spectral And Total Irradiance RE-construction model-Space era), available at: http://www2.mps.mpg.de/projects/sun-climate/data.html (http://www2.mps.mpg.de/projects/sun-climate/data.html)
- SATIRE-M (Spectral And Total Irradiance RE-construction model - Millennia), available at: http://www2.mps.mpg.de/projects (http://www2.mps.mpg.de/projects)

The total solar irradiance values range from 1356.656 to 1363.525. We use a feature scaling technique, also known as data normalization, to normalize the range of data to increase the cohesion of the TSI values. Specifically, we use the min-max normalization that is calculated as follows:

$$\hat{v}_i = \frac{v_i - \min(S)}{\max(S) - \min(S)}$$

where $\hat{v}_i$(v_i, respectively) is the normalized value (actualvalue, respectively) at time pointi, and S represents the input data set. The normalized TSI values range from 0 to 1.
Figure 2 shows the data scaling while keeping the same data shape as the original data as seen in Figure 1.

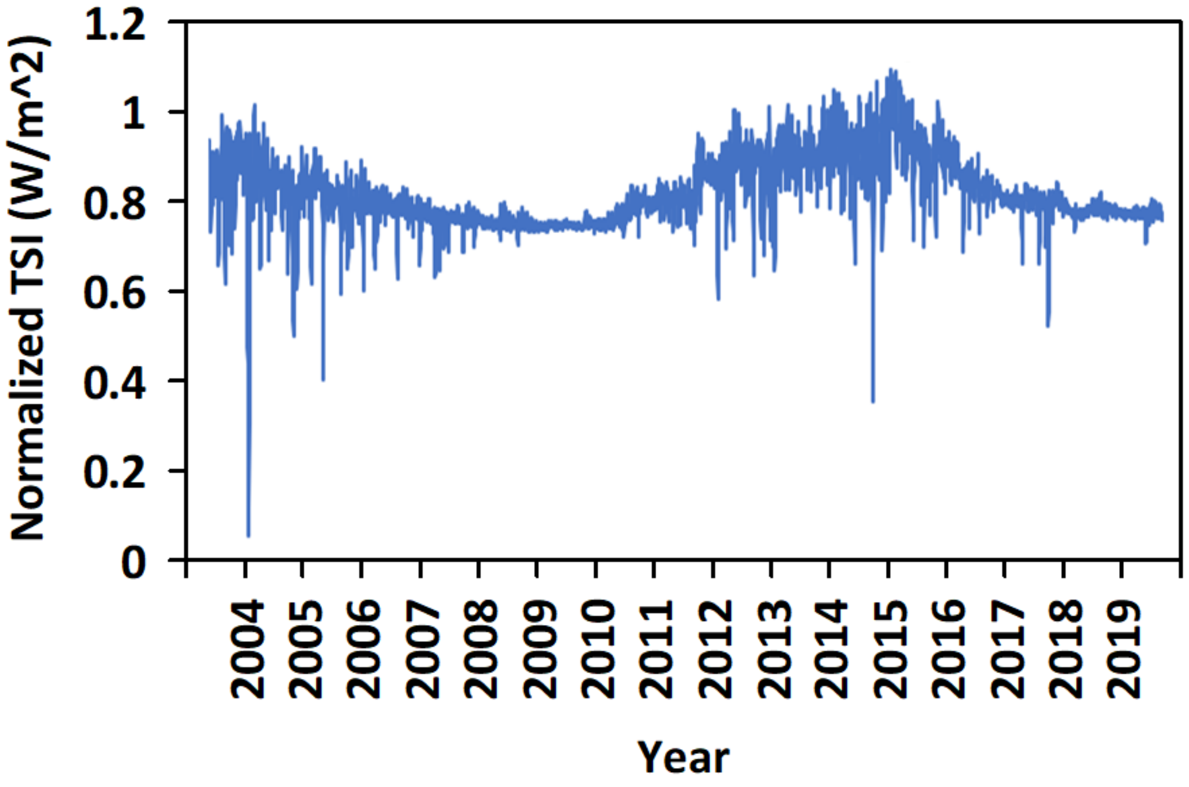


Figure 1: SORCE total solar irradiance (TSI) data from 2003 to the present (2021) with Data Scaling.

Binder

This notebook is Binder enabled and can be run on [mybinder.org (https://mybinder.org/)](https://mybinder.org/) by using the image link below:

launch binder [(https://mybinder.org/v2/gh/ccsc-tools/TSInet-irradiance-reconstruction/HEAD?labpath=YA_01_ReconstructionOfTSIbyDeepLearning.ipynb)](https://mybinder.org/v2/gh/ccsc-tools/TSInet-irradiance-reconstruction/HEAD?labpath=YA_01_ReconstructionOfTSIbyDeepLearning.ipynb)

Please note that starting Binder might take some time to create and start the image.

TSInet Workflow and Results

Data Preparation and Loading

The required files are included in the root folder of the github which includes all training and test data sets required to run the run the notebook. The files are loaded and used during the testing and training process.

Reconstructing with Pretrained Models

There are default and pretrained models that can be used to predict without running your own trained model. The models_directory is set to default_models which uses all pretrained algorithms.

In [5]:
```python
#Test default models.
from tsinet_reconstructor import *
dataset_name='TCTE'
model_directory='default_model'
print('Testing the default model with data set:', dataset_name)
args={}
result_file_name=r'results.txt'
args['dataset_name'] = dataset_name
args['model_dir']  = model_directory
args['result_file_name']=result_file_name
print('Process the require arguments..')
process_args(args)

reconstruct_tsinet()
```

```
Testing the default model with data set: TCTE
Process the require arguments..
model_dir: default_model

Please wait while reconstructing...
Loading model file: default_model\tsinet_model
Reconstruction will be performed using the model file: default_model\tsinet_m
odel
1/1 [==============================] - 1s 928ms/step
Processing with k steps = 5
1/1 [==============================] - 0s 17ms/step
1/1 [==============================] - 0s 19ms/step
1/1 [==============================] - 0s 18ms/step
1/1 [==============================] - 0s 18ms/step
1/1 [==============================] - 0s 19ms/step
1/1 [==============================] - 0s 18ms/step
1/1 [==============================] - 0s 19ms/step
1/1 [==============================] - 0s 18ms/step
1/1 [==============================] - 0s 18ms/step
```

Plotting the Pretrained Models Results

The prediction result can be plotted using the function plot_figures. It uses the results produced by the model from the "default_results" directory.

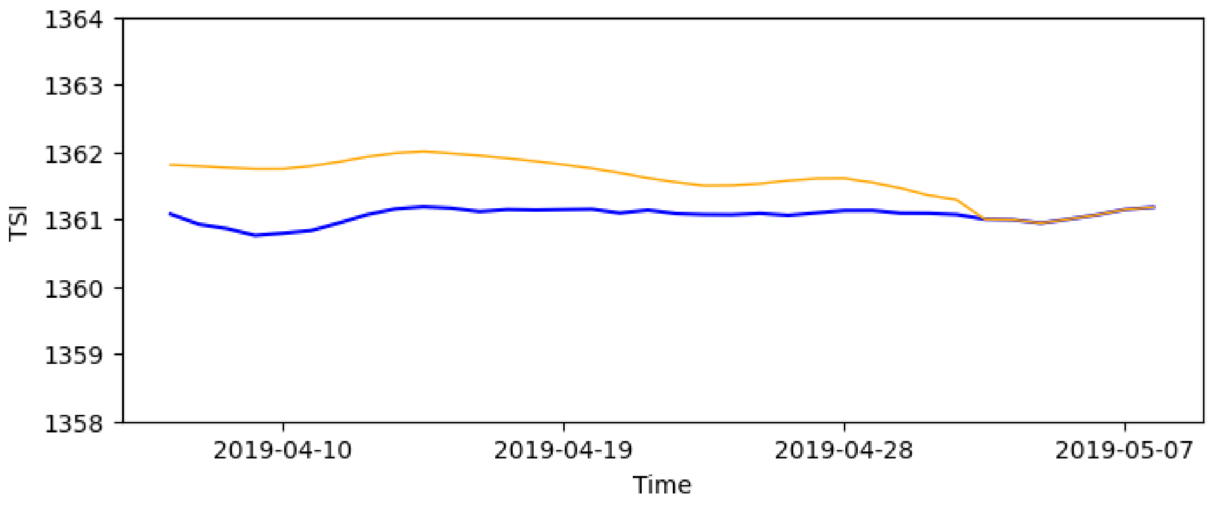
In [6]:
```
from tsinet_utils import plot_figures

dataset_name='TCTE'
result_file_name = 'results/tsinet_result_' + dataset_name + '.csv'
print('Plotting figures for default models results for TSI reconstruction of
plot_figures(result_file_name)

```

```
Plotting figures for default models results for TSI reconstruction of data set:
TCTE


*NOTE: the figure scale is large compared to the difference between the observe
d and predicted values, therefore you may see a big difference, but in actualit
y, the absolute difference between the values are very small.
```



TSInet Model Training and Testing Example

TSInet Model Training with Sample Data

Here, we show how to train the model with sample data example. In this exmaple, we show how to train the model.

In [8]:
```python
#Training for  sample data.
from tsinet_train import *
print('Training custom model')
training_file='train_data/SORCE_TSI.csv'
epochs=10
args={}
args['training_file'] = training_file
args['epochs'] = epochs

print('Process the require arguments..')
process_args(args)

train_tsinet()

```

```
Training custom model
Process the require arguments..
Epoch 1/10
356/356 [==============================] - 3s 5ms/step - loss: 0.0109
Epoch 2/10
356/356 [==============================] - 2s 5ms/step - loss: 9.1066e-04
Epoch 3/10
356/356 [==============================] - 2s 5ms/step - loss: 6.6391e-04
Epoch 4/10
356/356 [==============================] - 2s 5ms/step - loss: 5.1002e-04
Epoch 5/10
356/356 [==============================] - 2s 5ms/step - loss: 4.5365e-04
Epoch 6/10
356/356 [==============================] - 2s 5ms/step - loss: 4.2549e-04
Epoch 7/10
356/356 [==============================] - 2s 5ms/step - loss: 4.1997e-04
Epoch 8/10
356/356 [==============================] - 2s 5ms/step - loss: 3.5968e-04
Epoch 9/10
356/356 [==============================] - 2s 5ms/step - loss: 4.2654e-04
Epoch 10/10
356/356 [==============================] - 2s 5ms/step - loss: 4.6124e-04
Saving the TSInet model
```

```
WARNING:absl:Found untraced functions such as _jit_compiled_convolution_op, _ji
t_compiled_convolution_op, _jit_compiled_convolution_op while saving (showing 3
of 3). These functions will not be directly callable after loading.
```

```
The model is saved in the models directory
Finished TSInet training...
```

Reconstructing with Your Trained TSInet Model

To reconstruct the testing data using the model you trained above, use the following settings. Note: this is training job is only an example that uses less training proceses, and epochs, therefore the results and performance metrics may not be comparable to fully developed pretrained and default models.

In [9]:

```
#Test default models.
from tsinet_reconstructor import *
import os
dataset_name='TCTE'
model_directory='models'
result_file_name = 'results' +os.sep +'sinet_result_'+dataset_name+'.csv'
print('Testing the custom model with data set:', dataset_name)
args={}

args['dataset_name'] = dataset_name
args['model_dir']  = model_directory
args['result_file_name'] = result_file_name

print('Process the require arguments..')
process_args(args)

reconstruct_tsinet()
```

```
Testing the custom model with data set: TCTE
Process the require arguments..
model_dir: models

Please wait while reconstructing...
Loading model file: models\tsinet_model
Reconstruction will be performed using the model file: models\tsinet_model
1/1 [==============================] - 0s 250ms/step
Processing with k steps = 5
1/1 [==============================] - 0s 18ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 17ms/step
1/1 [==============================] - 0s 18ms/step
1/1 [==============================] - 0s 18ms/step
1/1 [==============================] - 0s 17ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 18ms/step
1/1 [==============================] - 0s 18ms/step
```

Plotting the Results for Your Trained Model

The prediction result can be plotted using the function plot_figures as shown in the following example.

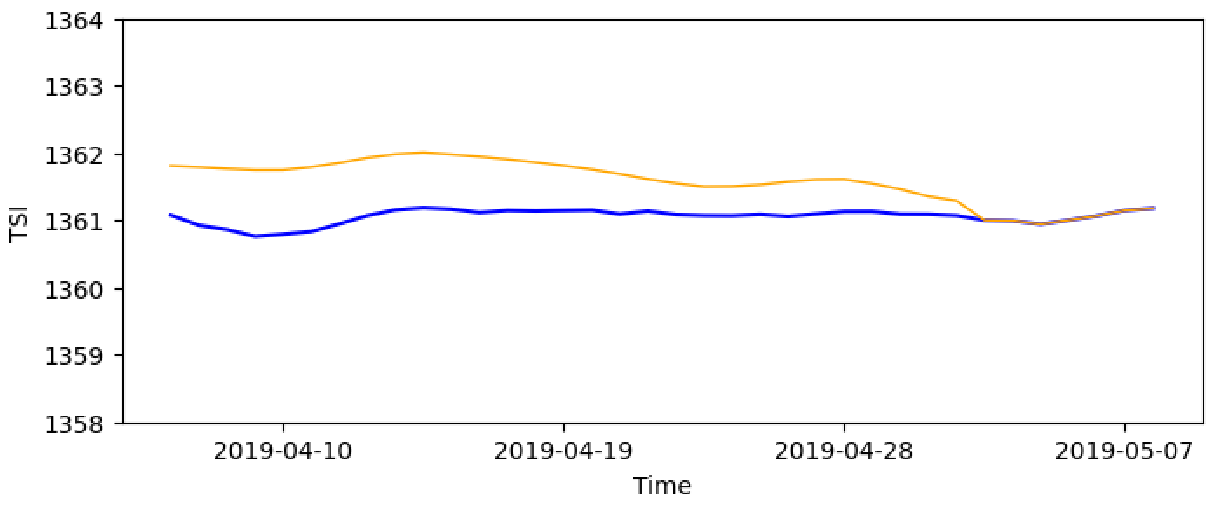
In [10]:
```
from tsinet_utils import plot_figures

dataset_name='TCTE'
result_file_name = 'results/tsinet_result_' + dataset_name + '.csv'
print('Plotting figures for default models results for TSI reconstruction of
plot_figures(result_file_name)

```

```
Plotting figures for default models results for TSI reconstruction of data set:
TCTE


*NOTE: the figure scale is large compared to the difference between the observe
d and predicted values, therefore you may see a big difference, but in actualit
y, the absolute difference between the values are very small.
```



Timing

Please note that the execution time in mybinder varies based on the availability of resources. The average time to run the notebook is 10-15 minutes, but it could be more.

Conclusions

The Earth’s primary source of energy is the radiant energyfrom the Sun. This energy is known as solar irradiance, ortotal solar irradiance (TSI) when all of the radiation is mea-sured. The changes in solar irradiance have a significant im-pact on Earths’ atmosphere and climate. Therefore, studying and reconstructing solar irradiance is crucial in solar physics. Existing methods for solar irradiance reconstruction are all based on physics-based models. In this paper, we presented the first deep learning method(TSInet) for reconstructing total solar irradiance (TSI). Experimental results showed that results from our TSInet agree well with those from the physics-based models. When compared to closely related machine learning methods, TSInet achieves the best performance among the methods. TSInet does not depend on physics properties such as proxies, and hence it can be extended back at times when proxies were not available. We demonstrated here that TSInet is able toreconstruct TSI for more than 9 millennia.

References

1. Reconstruction of Total Solar Irradiance by Deep Learning
   Yasser Abduallah, Jason T. L. Wang, Yucong Shen, Khalid A. Alobaid, Serena Criscuoli, and Haimin Wan
   https://doi.org/10.32473/flairs.v34i1.128356 (https://doi.org/10.32473/flairs.v34i1.128356)
2. Where does Earth's atmosphere get its energy?
   Kren, A. C. and Pilewskie, P. and Coddington, O.
   https://doi.org/10.1051/swsc/2017007 (https://doi.org/10.1051/swsc/2017007)
3. Solar irradiance variability: A six-year comparison between SORCE observations and the SATIRE model
   Ball, W. T. and Unruh, Y. C. and Krivova, N. A. and Solanki, S. and Harder, J. W.
   https://doi.org/10.1051/0004-6361/201016189 (https://doi.org/10.1051/0004-6361/201016189)
4. Solar activity over nine millennia: A consistent multi-proxy reconstruction
   Wu, C. J. and Usoskin, I. G. and Krivova, N. and Kovaltsov, G. A. and Baroni, M. and Bard, E. and Solanki, S.K.
   https://doi.org/10.1051/0004-6361/201731892 (https://doi.org/10.1051/0004-6361/201731892)

In [ ]:
```

```