Manuscript Title

Authors

- **Charlotte Cherry**
  · ⓞ [ccherry2](#)
  Department of Civil Engineering, University of Illinois Urbana-Champaign

- **Joyce Yang**

  Department of Civil Engineering, University of Illinois Urbana-Champaign

- **Yiwen Zhang**
  · ⓞ [Yiwen-Zhang97](#)
  Department of Civil Engineering, University of Illinois Urbana-Champaign

Abstract

II. Introduction

Urban areas take up a relatively small percentage of the Earth's land cover but have disproportionately large impacts on the climate and on humans. They are major drivers of emissions and climate change, serve as economic and social centers around the world, and house most of the human population. However, the very nature of their small physical footprint makes it challenging to study their impact on humans and the environment accurately and comprehensively.

To grapple with uncertainty and develop climate change mitigation and/or adaptation strategies, it is crucial for policymakers and planners for urban areas to understand different climate projections and scenarios, as well as urban-specific dynamics. Earth Systems Models (ESMs) are complex mathematical models that produce climate projections. They represent physical processes in the atmosphere, ocean, cryosphere, biogeochemical cycling in terrestrial and marine ecosystems, and interactions and feedbacks between these domains. These models are highly computationally demanding – taking long amounts of time and storage capacity to run over timescales of hundreds of years. This can make ESMs impractical for many uses, particularly policy analyses. Interested users may not have the capacity to utilize GCMs on a typical computer or reasonable budget.

Currently, most state-of-the-art ESMs used today for climate change projections do not explicitly parameterize urban areas, largely due to their small area. While this does not significantly impact the quality of regular or large-scale studies, this lack of explicit parameterization limits our ability to adequately capture unique urban characteristics and dynamics. The Community Earth Systems Model (CESM) is one of the few state-of-the-art ESMs that explicitly parameterizes urban areas, with the latest version even distinguishing between three separate urban density classes. Most quantitative attributions have been typically done for non-urban surfaces, but effective development decisions and local actions to manage risks rely on robust urban climate projections. This is the motivation for us to use CESM, which has a representation of urban areas, to build a location dependent emulator, and apply it to other ESMs in order to get their urban temperature responses.

While CESM provides the advantage of explicit urban parameterization, it does still require significant supercomputing resources, which may limit its usefulness. Therefore, there is a desire to use artificial intelligence to reduce this load. A climate emulator could achieve this by statistically replicating the nonlinear behavior of ESMs more quickly and with less computing power. This project will use machine learning methods to develop a model that can emulate urban temperatures (using other atmospheric forcing variables), where the risk of heat waves in the future could have the greatest negative impacts on human health. This model will be loosely based off of the conceptual framework presented by Zhao et al (2020) - "Global multi-model projections of local urban climates" (currently in press). Urban areas in this dataset refers loosely to areas where people live (i.e., not oceans or uninhabitable areas) but they do not exclusively correspond to cities.

III. Methods

A. Exploratory Data Analysis Findings

Exploratory data analysis was performed on the training set to understand the dataset as well as the key variables included and their associated patterns. We also wanted to identify relationships among variables, which we believe will help us choose the best predictors for our model.

The training dataset contains 486 columns and 392118 rows. It includes the monthly mean of each variable from 2015 to 2100, with one year selected per decade, totaling 108 months. The longitude resolution is 288 pixels, and the latitude resolution is 192 pixels. Only the populated land area is retained. Therefore, each row represents variables in each pixel for each time step. The large number of rows indicates that we have enough samples to train the model. For columns, in addition to the target variable (urban 2-m air temperature or "TSA") that we want to predict, we still have 485 variables that could be used as inputs to the model. Among those, some are merely descriptive information which do not provide useful clues for predicting TSA, such as current day or nstep, so they can be safely excluded.

We wanted to further narrow down the range of variables to use as inputs, which was done by identifying the correlation between potential independent variables and the target variable. It was found that variables that were highly correlated with TSA were also temperature variables, many of which were interdependent and would not add much value to the model due to multicollinearity. Those variables are listed below.

- TSA_ICE = 2m air temperature (ice land units only)
- TBOT = atmospheric air temperature
- TG_ICE = ground temperature (ice land units only)
- TV = vegetation temperature
- TSKIN = skin temperature
- THBOT = atmospheric air potential temperature
- TG = ground temperature
- TREFMNAV = daily minimum of average 2-m temperature
- TSL = temperature of near-surface soil layer
- TSOI_10CM = soil temperature in top 10cm of soil
- TREFMXAV = daily maximum of average 2-m temperature
- TH2OSFC = surface water temperature
- TBUILD = internal urban building air temperature
- WA = water in the unconfined aquifer
- ZBOT = atmospheric reference height
- Vcmx25Z = canopy profile of vcmax25
- TOPO_COL_ICE = column-level topographic height
- ZWT_PERCH = perched water table depth

Therefore, when preparing data for our model we would need to filter the predictors to only include forcing variables. We also thought that adding interaction terms between multiple variables (e.g. feature crossings of spatial and time variables such as RAIN*month) in the model would be helpful, and this might be achieved by letting the model learn the relationships by itself without manually specifying them.

The next step was to delve into some valuable variables by visualizing them both spatially and temporally to understand the range and distribution, as well as identify anomalous values.

 1. TSA

Since TSA is our target output, we focused much of our effort on analyzing it.

Some NAN values of TSA from 2035-08 and 2035-09 were cleaned up. Time series plots of global monthly mean temperature in urban areas showed seasonal fluctuations. The highest TSA appeared in JJA every year, which might be because there are more urban areas in the northern hemisphere where summer is in JJA. The figure below shows the high and low temperature for each year in our

time series, and the flat period in between represents the years where we do not have data (we have data for 1 out of every 10 years).

Figure: time series plot of global monthly average TSA in urban areas

When these data are considered throughout the full timeline, there was an upward trend in the maximum, mean, and minimum annual temperature over the time range of projections. This climate change trend is something we would want to be able to capture in our model.

Figure: maximum, mean and minimum annual temperature trend

The standard deviation of TSA over time was also assessed, which tended to be higher in DJF than in JJA, and the magnitude did not seem to change significantly. This suggested that the underlying drivers of TSA may remain steady, even as the overall average increases.

Figure: time series plot of the standard deviation of global monthly average TSA in urban areas

We also examined these temperature data spatially to see how the data vary globally. As expected, the figure below shows the seasonal patterns between the northern and southern hemispheres, as well as higher temperatures near the equator. These spatial patterns of temperature in 2015, 2055 and 2095 remain similar but it is clear that warming is occurring. In particular, the Indian subcontinent appears to become warmer in the series of maps. It is also important to note that the majority of our data points are located in the northern hemisphere.

Figure: Maps of TSA in January and June of 2015, 2055 and 2095

2. Other temperature variables

We examined other temperature variables that were highly correlated with TSA to verify the hypothesis of multicollinearity. Variables explored included vegetation temperature (TV), surface water temperature (TH2OSFC), internal urban building air temperature (TBUILD).

Vegetation temperature and surface water temperature had a similar spatial distribution to TSA, demonstrating high multicollinearity. The internal building air temperature had a very different spatial distribution compared to the vegetation temperature, surface water temperature, and TSA. This was likely because it is not an atmospheric variable and is probably strongly influenced by human decision-making.

Figure: Maps of TV, TH2OSFC and TBUILD in January of 2015

3. Some atmospheric forcing variables

Some atmospheric forcing variables were also worth exploring because they are inputs to every earth system model and provide a lot of information about TSA prediction, although they are not as correlated to TSA as other temperature variables. The variables chosen were according to Zhao et al (2020). The descriptions and units of these variables are listed below. * FSDS (W/m^2): atmospheric incident solar radiation * FLDS (W/m^2): atmospheric longwave radiation * RAIN (mm/s): atmospheric rain, after rain/snow * TBOT (K): atmospheric air temperature * PBOT (Pa): atmospheric pressure at surface * QBOT (kg/kg): atmospheric specific humidity * U10 (m/s): 10-m wind The correlation matrix showed that although these variables were not as highly correlated with TSA, they were still fairly correlated, especially for TBOT and FLDS.

Figure: correlation matrix of TSA and atmospheric forcing variables

We compared the spatial pattern of these variables to that of TSA, taking January 2015 as an example. The patterns of TBOT, FSDS, FLDS and QBOT were similar to those of TSA, whose values were larger near the equator and decreased with the increase of latitude. Other variables (RAIN, PBOT, U10) did not have clear spatial patterns, which made sense since they were not related to geospatial attributes.

Figure: Maps of TSA, TBOT, FSDS, FLDS, RAIN, PBOT, U10 and QBOT in January of 2015

Overall, we were able to gather useful information from the exploratory data analysis. The data quality was improved by identifying and removing outliers in the dataset. We learned how the temperature changed over the course of the century. We will try different combinations of variables with different algorithms to see how to achieve the best accuracy with the least amount of computing resources.

B. Preprocessing Data for Model

C. Model Testing

1. Neural Network

2. Random Forest (Selection of Variables)

3. Random Forest (Minimized Variables)

IV. Results

A. Overview

Overall, the random forest models tended to do better than the neural networks that we tried, however, model performance (quantified as mean absolute error) was highly dependent on feature selection. We had to keep in mind two main goals as we developed our model. On one hand, we wanted to reduce the mean absolute error, so the model could predict urban temperature with the greatest accuracy possible. This would improve the usefulness of the model for urban planners, policymakers, and other stakeholders who could use results from this emulator. However, the main motivation behind this project was to build a model that would be easily adapted to run on other earth systems models participating in CMIP6. For this emulator to be useful for that, it would need to be easily adaptable to a wide variety of models. During model development, this manifests in decisions such as variable selection – using uncommon atmospheric variables would reduce the number of models this emulator could be used on, thus decreasing the adaptability. Considering the balance of these two goals (accuracy and adaptability), we drew from each of our individual models to synthesize the "best model." An outline of this model will be described below.

B. Preprocessing

We individually arrived at many common preprocessing steps, which will be incorporated into the "best model." We found that it was necessary to drop rows where TSA values were "NaN," in order to build the model at all. The time was converted into datetime format and processed in some way: either making a new column for the year or converting day and year information using trigonometric functions (to better represent their cyclical nature). The training data was split into consecutive training and validation samples, without random shuffling. This improves the robustness of the model by testing it on the last time intervals of the dataset. We also found that it was important to fill in NaN values in the feature columns – if we dropped all rows with any NaN values, this would result in a greatly diminished training dataset. We found that sklearn's "Simple Imputer" worked relatively well by

replacing missing values with the mean value for each feature. Finally, we normalized the data using the mean and standard deviation of the training data, since the features had a wide range of values and magnitudes. Further model development could investigate the reason for these TSA = NaN values. These values could be NaN due to issues in the conversion between netcdf and csv, an issue with the original file itself, issues in the extraction of urban grid cells, a characteristic of the simulation itself, etc. Based on the reason, these rows could either be dropped, or the missing values in the training data could be imputed.

C. Model Development

For our best model, we would recommend using a random forest machine learning model. Our hyperparameter tuning revealed that the number of trees (n_estimators) had the greatest impact on model error. We were able to achieve optimal results with 300 estimators. The model was not as sensitive to other hyperparameters, such as the maximum leaf nodes or minimum number of samples to split. Not bootstrapping did lead to poorer model performance. With greater computing resources, we would recommend trying a K-fold cross validation along with grid search to truly optimize hyperparameters. However, we found that most of the default hyperparameter settings worked relatively well for our model. Another aspect of model development that strongly influenced performance was variable selections. Our approaches ranged from using only seven common atmospheric variables to using over 40 of the feature columns. In order to improve the adaptability of the model, we did not use all of the features provided, since some of those were outputs of the model (similar to TSA, and likely having the same atmospheric drivers) and others are not commonly used in other earth systems models. In order to maximize the adaptability of the model while minimizing the error, we would recommend using 7-10 common atmospheric variables as features.

D. Future Development

In summary, our best model was a random forest model with 300 trees. We cleaned the data by dropping TSA values that were "NaN," converting time into datetime and extracting the year or transforming the year/month using trigonometric functions, splitting the training and validation data without shuffling (for hyperparameter tuning), filling in NaN values for the feature columns with their mean value, and normalizing the data by mean and standard deviation. We found that most of the default hyperparameter settings worked well for our model, with the exception of the number of trees. Finally, we experimented with a variety of feature column combinations. In the interest of balancing the accuracy and adaptability of the model, we would recommend using some variation of these variables: lat, lon, FSDS (atmospheric incident solar radiation), FLDS (atmospheric longwave radiation), RAIN (atmospheric rain), TBOT (atmospheric air temperature), PBOT (atmospheric pressure at surface), QBOT (atmospheric specific humidity), and U10 (10-m wind), and time feature(s) (year, or trigonometric transformations of year or month, to better represent the cyclical nature of time). By preprocessing and using this model architecture, we were able to achieve a RMSE of less than 0.20 degrees Kelvin. Further development of this model could focus on feature processing/extraction, or general model architecture. It would be interesting to incorporate more spatially explicit features, such as distance to coast, climate zone, etc. Lat/lon information could also be a more predictive feature with further manipulation, such as binning. We would recommend trying different methods of imputing NaN values, which could vary by the reason for the NaN values, and vary by the feature. Different selections and combinations of features could also be explored. After adjusting feature processing/extraction, model architecture and hyperparameters could be further developed. While there is room for further model improvement, our combined findings and recommended model provide a strong starting point for future work.

V. Discussion and Conclusion

A. Interpreting Results

The usefulness of this model and the level of accuracy that an RMSE of 0.20K represents depends on the desired application. For decision-making purposes, a RMSE of 0.20K is much lower than the uncertainty of future emissions, climate forcing/dynamics, and model structural uncertainty. For stakeholders/policymakers, this model would be most useful for predicting general patterns in urban temperature (both spatially and temporally), rather than for exact quantities. Additionally, it would be very useful in creating large ensemble projections of urban temperature under future climate change.

B. What We Learned

C. Use of Machine Learning for Earth Systems Model Emulation

References

Appendix 1: Variable Definitions

This table includes the full list of variables used in the models created and their definitions and units.

Table A1: Variable Names and Definitions

Variable	Unit	Meaning
TSA	*K*	*2m air temperature (target variable)*
ATM_TOPO	m	Atmospheric surface height
Day sin	day	Sine of day term
EFLX_LH_TOT	W/m^2	Total latent heat flux (+ to atm)
EFLX_LH_TOT_R	W/m^2	Rural total evaporation
ER	gC/m^2/s	Total ecosystem respiration, autotrophic + heterotrophic
ERRSEB	W/m^2	Surface energy conservation error
ERRSOL	W/m^2	Solar radiation conservation error
FCEV	W/m^2	Canopy evaporation
FCOV	unitless	Fractional impermeable area
FGEV	W/m^2	Ground evaporation
FH20SFC	unitless	Fraction of ground cover by surface water
FIRA	W/m^2	Net infrared (longwave) radiation
FIRA_R	W/m^2	Rural net infrared (longwave) radiation
FIRE	W/m^2	Emitted infrared (longwave) radiation
FIRE_ICE	W/m^2	Emitted infrared (longwave) radiation (ice landunits only)
FIRE_R	W/m^2	Rural emitted infrared (longwave) radiation
FLDS	W/m^2	Atmospheric longwave radiation
FSA	W/m^2	Absorbed solar radiation
FSDS	W/m^2	Atmospheric incident solar radiation
FSDSNI	W/m^2	Diffuse nir incident solar radiation
FSDSVD	W/m^2	Direct vis incident solar radiation
FSDSVI	W/m^2	Diffuse vis incident solar radiation
FSH_G	W/m^2	Sensible heat from ground
FSH_R	W/m^2	Rural sensible heat
FSN0	unitless	Fraction of ground covered by snow
FSR	W/m^2	Reflected solar radiation
FSRND	W/m^2	Direct nir reflected solar radiation
FSRNI	W/m^2	Diffuse nir reflected solar radiation
H20SFC	mm	Surface water depth
HIA	C	2m NWS Heat Index
HUMIDEX	C	2m Humidex
lat	degrees_north	Coordinate latitude
long	degrees_east	Coordinate longitude
PBOT	Pa	Atmospheric pressure at surface
Q0VER	mm/s	Surface runoff
Q2M	kg/kg	2m specific humidity
QBOT	kg/kg	Atmospheric specific humidity
QICE	mm/s	Ice growth/melt
QIRRIG	mm/s	Water added through irrigation
QRUNOFF	mm/s	Total liquid runoff not including correction for land use change
RAIN	mm/s	Atmospheric rain, after rain/snow
RH2M	%	2m relative humidity
SABG	W/m^2	Solar rad absorbed by ground
SNOWDP	m	Gridcell mean snow height
SOILWATER_10CM	kg/m2	Soil liquid water + ice in top 10cm of soil
SWBGT	C	2m Simplified Wetbulb Globe Temp
SWBGT_R	C	Rural 2m Simplified Wetbulb Globe Temp
SWBGT_U	C	Urban 2m Simplified Wetbulb Globe Temp
TBOT	K	Atmospheric air temperature
TBUILD	K	Internal urban building air temperature
TG	K	Ground temperature
TG_ICE	K	Ground temperature (ice landunits only)
TH20SFC	K	Surface water temperature
THBOT	K	Atmospheric air potential temperature
TOPO_COL_ICE	m	Column-level topographic height
TREFMNAV	K	Daily minimum of average 2m temperature
TREFMXAV	K	Daily maximum of average 2m temperature
TSA_ICE	K	2m air temperature (ice landunits only)
TSKIN	K	Skin temperature
TSL	K	Temperature of near-surface soil layer
TSOI_10CM	K	Soil temperature in top 10cm of soil