CO2 PREDICTOR FOR SUSTAINABLE BUSINESS

Deep Learning for Tracking Greenhouse Gas Emissions from Satellite Imagery

Addressing the issue

The burning of fossil fuels produces large amounts of carbon dioxide (CO_2), a major Greenhouse Gas (GHG) and a main driver of Climate Change. Quantifying GHG emissions is crucial for accurate predictions of climate effects and to enforce emission trading schemes.

To make business sustainable drilling down gas emissions is a necessity. For this, keeping track of CO_2 Emissions and have insights about future emissions becomes of utmost importance for the business as well as to the environment.

Business Impact

Quantifying GHG emissions from individual industrial sites typically requires the use of dedicated measuring devices that report detailed emission information, as may be mandated by environmental protection guidelines.

Such data driven insights may be used to enforce environmental protection regulations or pollutant certificate trading schemes.

Novelty

There has only been recent advancements in Data Science and AI/ML field. The used datasets are scarce and not all publicly available. So research in this field is at a budding stage yet. These topics are sensitive and can create major impacts. So, it's necessary to take care of Type-2 Errors.

Our approach

We plan to develop a deep learning approach able to simultaneously predict:

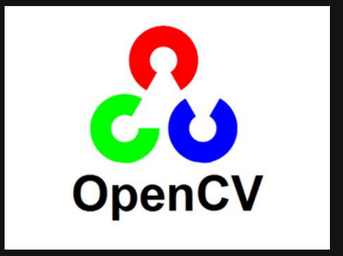
The pixel-area covered by plumes

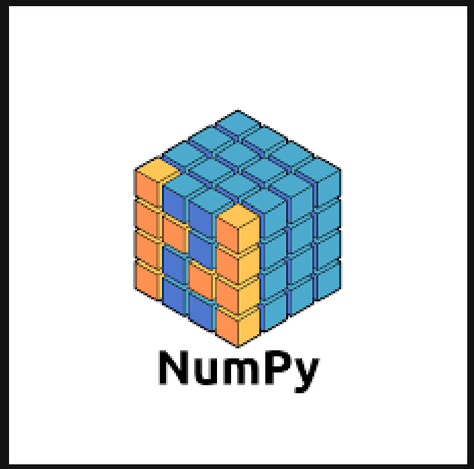
Area covered by the emissions

Future estimated Emissions

Our plan is to predict estimates for the rate at which CO2 will be emitted.

Technologies & Libraries Used

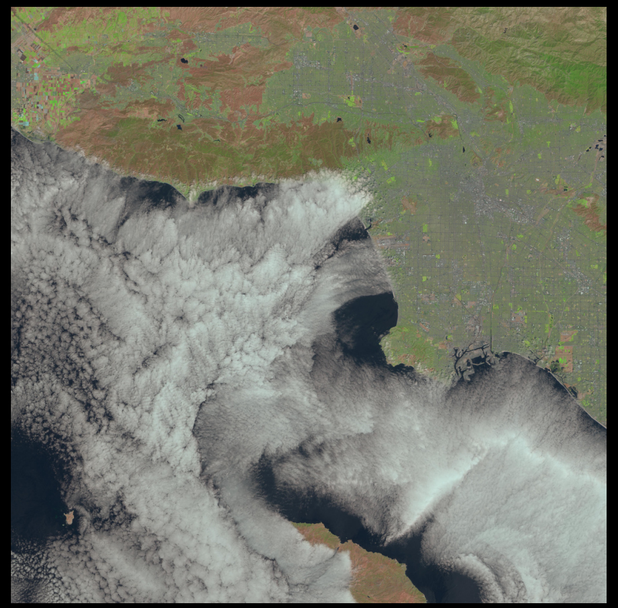


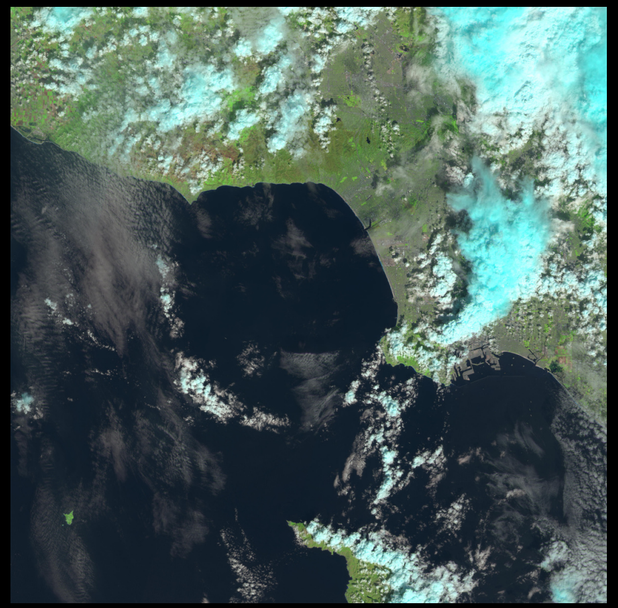


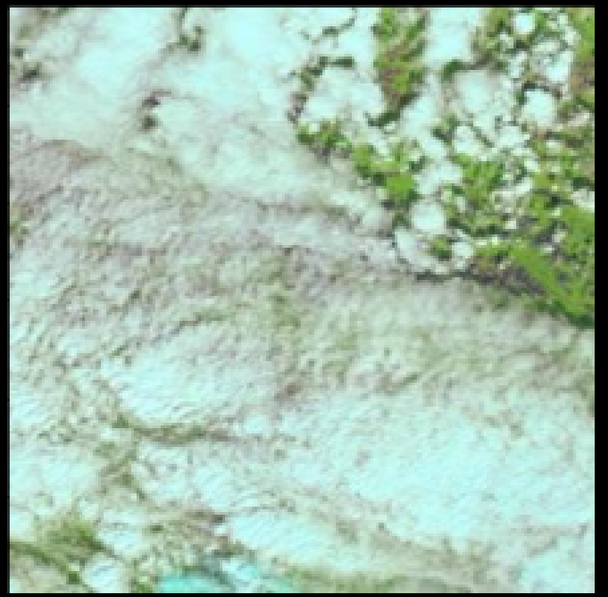
Proposed Methodology (1/5)

Dataset

1. **The Dataset was taken from a satellite name Sentinel 2.**
2. **225 Images of 5400px by 5400px from satellite were taken over 100 square km area of Los Angeles.**
3. **Dataset of 5 years from 2015-2020 were recorded.**



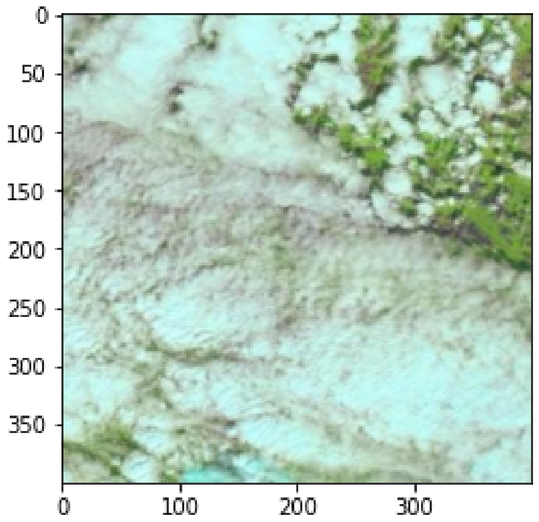


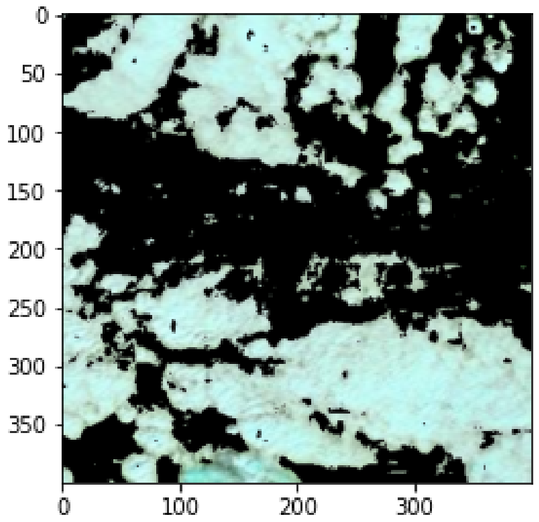


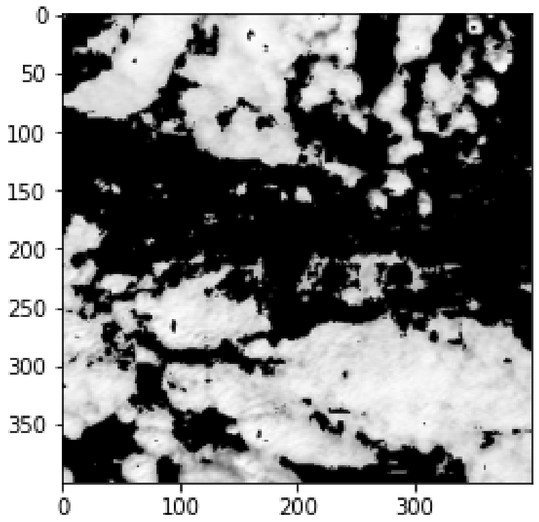
Proposed Methodology (2/5)

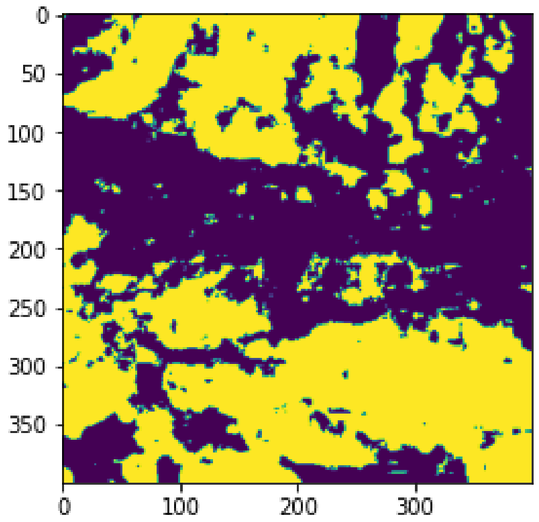
Image Preprocessing

1. **The images were loaded into the notebook via matplotlib's: plt.imread() function**
2. **All Images were then Resized into standard size 400 X 400 using cv2.resize()**
3. **RGB image was converted to HSV colors**
4. **A mask was applied on the images to segment gas emissions in the images**
5. **Image was represented in grayscale**









Proposed Methodology (3/5)

Sequential Model

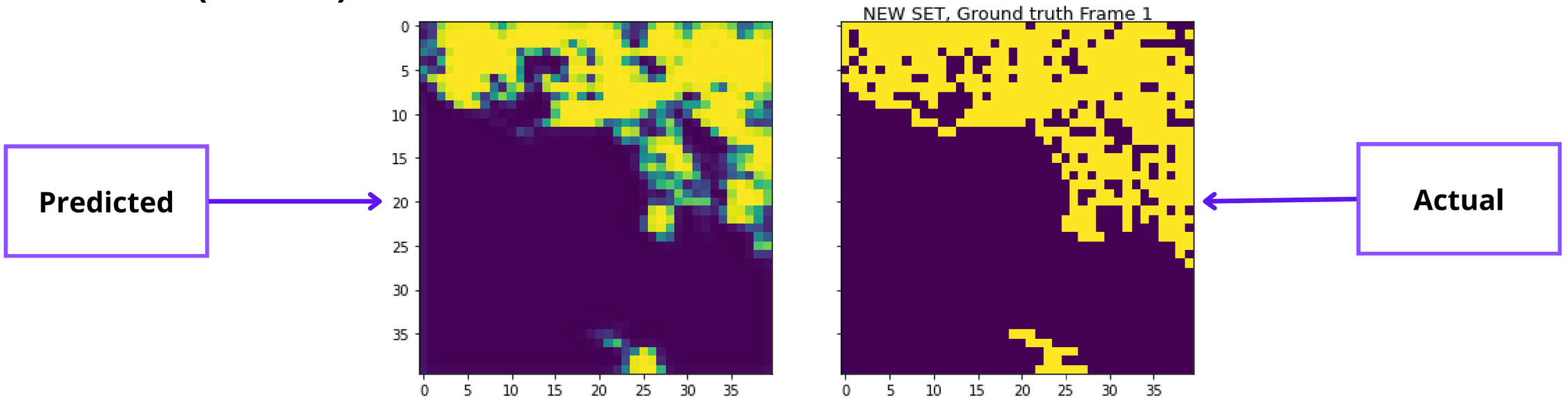
1. Keras Sequential model was initialised
2. A convLSTM2D was added with channels last type 4D Tensor input (samples, time, rows, cols, channels), same padding, and kernel of size 2x2
3. A BatchNormalization was added to make training faster
4. Steps 2, 3 were repeated to make the network dense but this time kernel size was 3x3 to extract most patterns.
5. return_sequences were set to True to avoid any loss or dimensionality reduction
6. Finally a Conv3D layer was added with kernel_size 3x3x3 and sigmoid activation function.

```
Model: "sequential"
_________________________________________________________________
 Layer (type)                Output Shape              Param #
=================================================================
 conv_lstm2d (ConvLSTM2D)    (None, 5, 40, 40, 1)      36

 batch_normalization (BatchN  (None, 5, 40, 40, 1)     4
 ormalization)

 conv_lstm2d_1 (ConvLSTM2D)  (None, 5, 40, 40, 1)      76

 batch_normalization_1 (Batc  (None, 5, 40, 40, 1)     4
 hNormalization)

 conv3d (Conv3D)             (None, 5, 40, 40, 1)      28

=================================================================
Total params: 148
Trainable params: 144
Non-trainable params: 4
_________________________________________________________________
```

Proposed Methodology (4/5)

Training & Testing

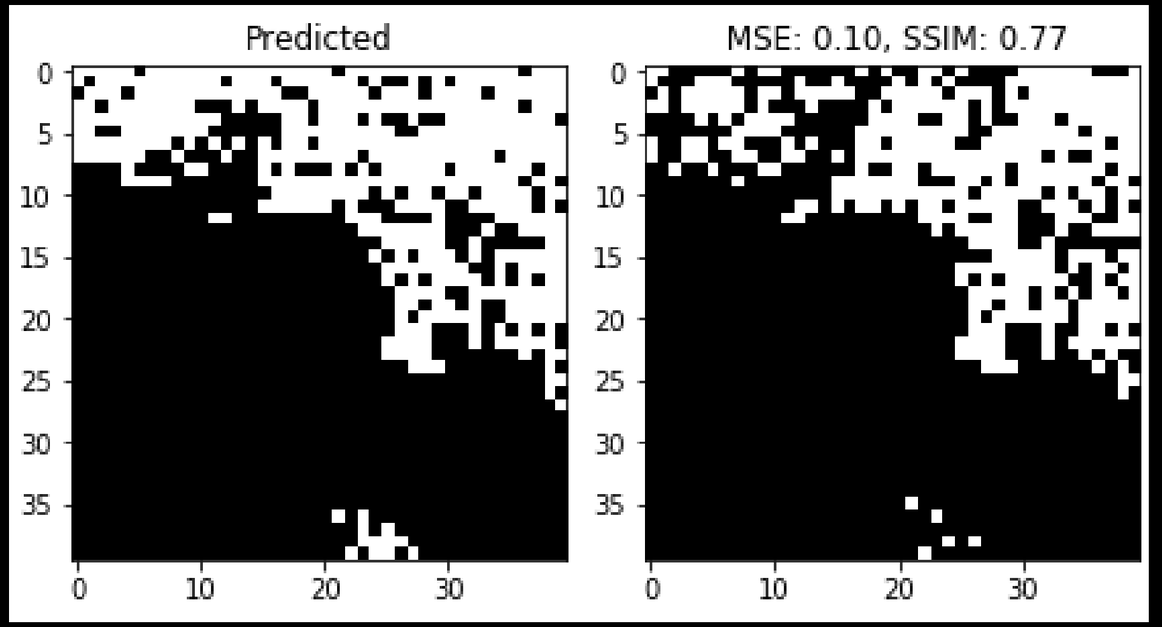
1. The sequential model was trained/fitted over the dataset for 50 epochs.
2. Latter one third part of the dataset was considered as test set.
3. The training set was split into 90% training and 10% validation set.
4. The training took about 5 minutes.
5. The trained model was then used to predict the Test set of images
6. Based on the previous year's pollution (train set), model was able to predict the future emissions (test set)



Proposed Methodology (5/5)

Error & Result Analysis

1. MAE, MSE were considered as loss functions. SGD, Adam were considered for optimizers.
2. Hyperparameter Tuning: After evaluating the model multiple times, Adam & MAE were selected.
3. MSE and SSIM were used for error analysis between the predicted and ground truth frames
4. MSE (Mean Squared Error) for calculating the error in prediction and SSIM (Structural similarity index) to assess the similarity of two images.
5. Mean error was about 0.27 overall
6. The model was able to predict for multiple gases as well. Bluish pixels represent NO2, yellowish patches are for CO2, and so on.



Future Scope

- For now the model limits itself to predict emissions.
- Time-lapse based prediction in the form of a video whose code is partially tried.
- In future we plan to upgrade this to do the same for other GHGs like NO, CO, Methane, etc.
- Also, we can try to quantify the amount of gases emitted and thereby have a clearer approach to the problem.

TEAM: HAX

Name	College	Year	Branch	Role
Rishabh Jain	Dwarkadas J. Sanghvi College of Engineering	II	Computer Engineering	Captain
Ayush Singh	Sardar Patel Institute of Technology	II	Information Technology	Member
Devang Shah	Dwarkadas J. Sanghvi College of Engineering	II	Computer Engineering	Member
Ayush Deshmukh	Dwarkadas J. Sanghvi College of Engineering	II	Computer Engineering	Member

Thank You...