INTERNATIONAL INSTITUTE OF
INFORMATION TECHNOLOGY

HYDERABAD

BTP Report - 1

Project Title: Geospatial Solutions for Smart (Power) Grid Implementations
Advisor: Rakesh Kiran Chand Sir
Context: BTP - 1 [2025]
Team Members:

- Jagankrishna
- Jakeer Hussain
- Kiran Gullapalli
- Susheel Krishna

1. Abstract

This project tackles the challenge of restricted access to real-time power grid data, a significant hurdle for research and development. The proposed solution utilizes publicly available night-time satellite imagery as a proxy to estimate power consumption patterns, based on the correlation between night light intensity and electricity usage. This approach also considers Land Use Land Cover (LULC) classifications for providing ideal power grid locations.

The primary objectives include evaluating the predictive capability of this alternative data source and developing a prototype system for forecasting future grid scenarios and offering management suggestions.

The methodology involved collecting and processing VIIRS night lights data from 2014-2022 for select cities. A ConvLSTM2D Deep Neural Network model was employed for spatio-temporal forecasting, predicting future night light intensity, and thereby power needs. Key outcomes include a prototype capable of multi-year forecasting and methods for identifying ideal power grid locations using light intensity, its temporal aspects, and LULC data. The project demonstrates a viable approach for power grid analysis and planning when direct data access is limited.

2. Introduction

Background and Motivation:

Smart power grids are crucial for modern infrastructure, enabling efficient and reliable electricity distribution. They integrate advanced technologies to monitor, control, and optimize energy flow, which is vital for meeting growing energy demands and integrating renewable energy sources. However, a significant challenge in advancing smart grid technology lies in the restricted access to real-time and historical power grid data. This data is often kept confidential due to privacy and security concerns, making it largely unavailable for civilian research and development. This lack of accessible data severely limits the ability of researchers and developers to create, test, and validate innovative solutions in the power grid domain.

Problem Statement:

The core problem addressed by this project is the critical need for alternative, publicly accessible data sources that can effectively serve as substitutes for actual grid data in research and analysis. Without such alternatives, progress in developing and refining smart grid technologies and management strategies is significantly hampered. This project seeks to bridge this data gap by exploring viable proxies that can enable continued innovation. Though the data might not be precise, we are using the data to find the results by using it as a proxy.

Project Objectives:

This project aims to address the above limitation by leveraging publicly available night-time satellite imagery (night lights data) to estimate and analyze power consumption trends. The primary objectives include:

- Using night lights data as a proxy for power usage estimation.
- Training a spatio-temporal deep learning model (ConvLSTM2D) to predict future power consumption patterns.
- Identifying ideal locations for future power grid expansion based on light intensity, land use, and urban density.
- Providing actionable visualizations and results to support regional or national-scale power planning efforts.

Scope of the Project:

This project utilizes night-time satellite imagery as a proxy for estimating power consumption patterns. Data from 2014 to 2022 for Washington DC, Nashvillie, and Delhi was collected and analyzed. The methodology involves spatio-temporal sequence forecasting using a Deep Neural Network model, specifically a ConvLSTM2D model. The project investigates methods for identifying ideal power grid locations by considering factors such as light intensities, the temporal aspect of light intensity, integrated LULC and light intensity, and GHSL building density. While challenges such as high-resolution data handling and inherent limitations of the night lights dataset were encountered, this project serves as a foundational prototype for future development. Future work could involve integrating additional data layers for renewable resource suggestions, training the model with a wider range of locations, incorporating actual smart grid data if accessible, and exploring applications in fault detection, demand consumption forecasting, and renewable energy integration.

3. Literature Review

Night-time lights (NTL) data have been widely used as a proxy for electricity consumption, especially in areas lacking accessible grid data. Bhattarai et al. (2023) reviewed over 200 models and found that data from newer sensors like VIIRS significantly improved estimation accuracy compared to older DMSP/OLS data. Their study also noted that model accuracy tends to decrease at finer spatial resolutions (e.g., city level), highlighting challenges in localized analysis. Despite these limitations, machine learning models, including neural networks, have shown promise in capturing the complex relationship between NTL intensity and power usage. Our project builds on these findings by using VIIRS-based NTL data and deep learning (ConvLSTM) to forecast power consumption trends, further enhanced with LULC and GHSL data for spatial planning. [ref]

4. Proposed Solution

To address the problem of limited access to real-world smart power grid data, our project proposes an alternative approach that leverages publicly available satellite imagery, specifically, night-time lights (NTL) data—as a proxy for electricity consumption patterns. The intensity of night lights is generally correlated with human activity and power usage, making it a suitable substitute for direct grid data.

We developed a deep learning-based spatio-temporal forecasting pipeline using ConvLSTM2D networks, which are capable of learning both spatial features from satellite images and temporal patterns across years. The model was trained on historical NTL images (2014–2022) for three regions: Washington DC, Nashville, and Delhi, and used to predict future images up to 2032. These predictions serve as indicators of future electricity usage patterns.To enhance the relevance of our insights, we tried with different auxiliary datasets such as:

- Land Use Land Cover (LULC): To restrict analysis to usable land areas.
- Global Human Settlement Layer (GHSL): To factor in building density and urbanization.

Additionally, we applied spatial clustering techniques (e.g., K-Means) on the brightest areas of the predicted NTL images to identify ideal locations for future power grid development. These suggestions were generated using four methods—based on raw brightness, temporal trends, LULC filters, and GHSL data.

Despite certain limitations such as image resolution and the inability to validate against real grid data, our solution demonstrates a scalable and accessible framework for power grid planning using open-source geospatial data and deep learning techniques.

5. Methodology

Data Acquisition:

The data acquisition for this project focused on obtaining and processing VIIRS Nighttime Day/Night Band (DNB) composite data to serve as a proxy for power consumption patterns.

Source - Night lights : The primary dataset used is the VIIRS Nighttime Day/Night Band Composites Version 1. This dataset is provided by the Earth Observation Group, Payne Institute for Public Policy, Colorado School of Mines.

- **Dataset Availability Period:** The dataset is available from 2012-04-01T00:00:00Z to 2025-03-01T00:00:00Z.
- **Earth Engine Snippet:** The Earth Engine snippet for accessing this dataset is `ee.ImageCollection("NOAA/VIIRS/DNB/MONTHLY_V1/VCMCFG")`.
- **Cadence:** The data has a monthly cadence. But we collected yearly data on average.
- **Pixel Size:** The pixel size is 463.83 meters.
- **Band Used:** The `avg_rad` band, representing average DNB radiance values in nanoWatts/sr/cm², was selected for this analysis.

Data was collected for the years 2014 to 2022. The study focused on three specific regions: Washington DC, Nashville, and Delhi. The data collection process involved the following steps, implemented using Google Earth Engine:

- **Initialize Years:** Define the start and end years for data collection (2014 to 2022).
- **Loop Through Years:** Iterate through each year within the specified range.
- **Define Date Range (Annual):** For each year, define the date range from the beginning of the year to the beginning of the next year.
- **Load & Filter ImageCollection ('avg_rad'):** Load the VIIRS DNB monthly image collection and filter it by the defined annual date range. The `avg_rad` band is then selected.

- **Calculate Yearly Average:** Compute the mean of the `avg_rad` images within the annual collection to get a single yearly average image. This image is then clipped to the defined geographical boundary of the study region (geometry).
- **Export Image to Drive (GeoTIFF):** The resulting yearly average image is exported to Google Drive as a GeoTIFF file. The export parameters include the image, a description, the destination folder, a file name prefix including the year, the study region, a specified scale (500 meters), maximum pixels, and the file format.

Source GHSL : The GHSL data provides information on the presence and density of built-up areas.

- **Dataset Name:** JRC Global Human Settlement Layer (GHSL) - Built-Up Surface, P2023A release, multitemporal.
- **Dataset Provider:** European Commission's Joint Research Centre (JRC).
- **Earth Engine Snippet:** The Earth Engine snippet for accessing this dataset is `JRC/GHSL/P2023A/GHS_BUILT_S`.
- **Dataset Availability:** While the code snippet uses years, the GHSL built-up surface dataset (GHS_BUILT_S) is typically provided for specific years (e.g., 1975, 1990, 2000, 2014, 2018). The code iterates through years from 2014 to 2030, suggesting the use of the most recent available data or a potential future release. The description of the dataset on the Earth Engine catalog will provide the exact availability.
- **Pixel Size:** The code uses a scale of 500 meters for export. The native resolution of the GHSL built-up surface data can vary, with the GHS_BUILT_S layer often having a resolution of approximately 30 meters. However, the export scale determines the resolution of the downloaded image.
- **Band Used:** The `built_surface` band was selected, which represents the built-up surface presence.

Data was collected for the years specified in the code (2014 to 2030), although the actual available years for the GHS_BUILT_S dataset should be confirmed from the dataset documentation. The process involved:

- **Initialize Years:** Define the start and end years for data collection (2014 to 2030).
- **Loop Through Years:** Iterate through each year.
- **Load Image:** Load the GHSL built-up surface image for the specific year.
- **Select Band:** Select the `built_surface` band.
- **Clip to Geometry:** Clip the selected band to the defined region of interest (geometry).
- **Export Image to Drive (GeoTIFF):** Export the clipped image to Google Drive as a GeoTIFF file. The export parameters include the image, a description, the destination folder, a file name prefix including the year, the region, a specified scale (500 meters), maximum pixels, and the file format.

Source - LULC : The MODIS LULC data provides classifications of land cover types.

- **Dataset Name:** MODIS Land Cover Type Yearly Global 500m.
- **Dataset Provider:** University of Maryland and Boston University.
- **Earth Engine Snippet:** The Earth Engine snippet for accessing this dataset is `MODIS/061/MCD12Q1`.

- **Dataset Availability:** The dataset is available yearly.
- **Cadence:** Yearly.
- **Pixel Size:** The native resolution is 500 meters, and the export scale is set to 500 meters.
- **Band Used:** The `LC_Type1` band was selected, which represents the IGBP (International Geosphere-Biosphere Programme) classification scheme.

Data was collected for the years 2014 to 2022. The process involved:

- **Define Region of Interest:** Define the geographical boundary for data collection (geometry).
- **Load Dataset:** Load the MODIS LULC image collection and select the `LC_Type1` band.
- **Loop Through Years:** Iterate through each year from 2014 to 2022.
- **Filter by Year:** Filter the dataset to get the image for the specific year. Since it's a yearly product, `.first()` is used after filtering.
- **Clip to Geometry:** Clip the yearly image to the defined region of interest.
- **Export Image to Drive (GeoTIFF):** Export the clipped image to Google Drive as a GeoTIFF file. The export parameters include the image, a description, the destination folder, a file name prefix including the year, the region, a specified scale (500 meters), CRS (EPSG:4326), and maximum pixels.

```
var startYear = 2014;
var endYear = 2022;

for (var year = startYear; year <= endYear; year++) {
  var startDate = ee.Date.fromYMD(year, 1, 1);
  var endDate = ee.Date.fromYMD(year + 1, 1, 1);

  var dataset = ee.ImageCollection('NOAA/VIIRS/DNB/MONTHLY_V1/VCMCFG')
                .filterDate(startDate, endDate)
                .select('avg_rad');

  var yearlyAvg = dataset.mean().clip(geometry);

  Export.image.toDrive({
    image: yearlyAvg,
    description: 'NTL_' + year,
    folder: 'NTL_01_Washington',
    fileNamePrefix: 'NTL_' + year,
    region: geometry, // Note: Geometry will be specific to the region
    scale: 500,
    maxPixels: 1e13,
    fileFormat: 'GeoTIFF'
  });
}
```

```
var startYear = 2014;
var endYear = 2030;

for (var year = startYear; year <= endYear; year++) {
  var image = ee.Image('JRC/GHSL/P2023A/GHS_BUILT_S/' + year.toString());
  var builtSurface = image.select('built_surface');

  // Clip the image to the region of interest (geometry)
  var clippedImage = builtSurface.clip(geometry);

  Export.image.toDrive({
    image: clippedImage,
    description: 'GHSL_' + year,
    folder: 'GHSL_01_Washington', // Note: Folder name may vary per region
    fileNamePrefix: 'GHSL_' + year,
    region: geometry, // Note: Geometry will be specific to the region
    scale: 500,
    maxPixels: 1e13,
    fileFormat: 'GeoTIFF'
  });
}
```

```
// Define your region of interest
var geometry = [[][][][]]

// Load the MODIS land cover dataset
var dataset = ee.ImageCollection('MODIS/061/MCD12Q1').select('LC_Type1');

// Loop through each year from 2014 to 2022
for (var year = 2014; year <= 2022; year++) {
  // Filter the dataset for the specific year
  var image = dataset
    .filter(ee.Filter.calendarRange(year, year, 'year'))
    .first()
    .clip(geometry);  // Clip to your area of interest

  // Export the image
  Export.image.toDrive({
    image: image,
    description: 'MODIS_LULC_' + year,
    folder: 'GEE_LULC_Exports_2', // Note: Folder name may vary per region
    fileNamePrefix: 'MODIS_LULC_' + year,
    region: geometry,
    scale: 500,  // MODIS native resolution
    crs: 'EPSG:4326',
    maxPixels: 1e13
  });
}
```

Fig : code snippets to download datasets from google earth engine

Data Preprocessing (NTL Prediction Pipeline):

The acquired GeoTIFF files were processed through a pipeline to prepare them for the machine learning model. This pipeline included format conversion, resizing, and normalization.

The GeoTIFF files obtained from Google Earth Engine were first converted to PNG format and resized to a specific target resolution. This was necessary to standardize the input format and spatial dimensions for the subsequent machine learning steps.

The conversion process, as implemented in the `converter.py` script, handles different datasets (NTL, LULC, GHSL). For Nighttime Lights (NTL) data, the script reads the GeoTIFF, clips values to a defined range (0-63), normalizes them to a 0-255 grayscale range, and enhances brightness before saving as a PNG. For LULC data, it reads the GeoTIFF and creates a binary image highlighting a specific land cover class (value 13, potentially representing urban areas) as white (255) and others as black (0), saving the result as a PNG. The GHSL conversion involves reading the GeoTIFF, handling no-data values, cropping to the bounding box of valid data, normalizing the valid pixels, and saving as a grayscale PNG.

After conversion, the PNG images were resized to a target resolution of (317, 229) pixels using the `res_decreaser.py` script. This script iterates through the converted PNG files in a specified input folder, opens each image, resizes it using the LANCZOS resampling filter for high-quality downsampling, and saves the resized image to an output folder.

The resized grayscale PNG images of Nighttime Lights were then fed into a data pipeline for preparing them for the machine learning model.

- **Importing Libraries:** Necessary libraries such as `os`, `numpy`, `cv2`, `tensorflow`, `sklearn`, and `matplotlib` were imported.
- **Enabling Mixed Precision:** Mixed precision was enabled for potentially faster training on compatible GPUs using `set_global_policy('mixed_float16')`.
- **Defining Image and Sequence Parameters:** Parameters such as the image size (`IMG_SIZE` = (229, 317)) and the sequence length (`SEQUENCE_LENGTH` = 3, representing the number of past images used for prediction) were defined.
- **Data Loading:**
    - The process involved looping over folders in a base directory, where each folder likely contained the annual NTL images for a specific geographical location.
    - For each image file (assumed to be PNG), the image was loaded in grayscale using OpenCV (`cv2.imread`) and its pixel values were normalized to the range [0, 1] by dividing by 255.0.
- **Creating Sequences:**
    - The core idea is to create sequences of past images as input (X) to predict the subsequent image (y).
    - For a defined `SEQUENCE_LENGTH`, sequences were created by taking a consecutive set of images of that length as input (X) and the image immediately following the sequence as the target (y). For example, using images from Year 1, 2, and 3 to predict Year 4.
    - The loaded images for all locations were combined to form the complete dataset of input sequences (X) and corresponding target images (y).
    - The dimensions of the input sequences (X) and target images (y) were expanded to include a channel dimension, which is required by the ConvLSTM2D layer in the model.
- **Splitting and Batching:**
    - The combined data was split into training and testing sets using `train_test_split` from `sklearn.model_selection`. A test size of 20% was used, and the data was shuffled.
    - TensorFlow datasets were created from the training and testing sets with a specified batch size (e.g., 16) for efficient model training.
- The data loading and sequence creation process is implemented in the [`kaggle script`] script through functions like `load_and_preprocess_data`, `create_sequences`, and `load_all_places_data`.

Machine Learning Approach for NTL Prediction:

- **Problem Type:** The problem of predicting future Nighttime Lights images based on historical sequences was framed as a spatio-temporal sequence forecasting problem.
- **Method:** A Deep Neural Network model, specifically designed to handle both spatial and temporal dependencies in image data, was employed. The model learns patterns within each image (spatial features) and how these patterns evolve over time across the image sequence (temporal patterns).
- **Model Details:** The chosen model architecture is based on the ConvLSTM2D layer, which is particularly well-suited for this type of spatio-temporal forecasting task.
    - **ConvLSTM2D Model**: This layer integrates convolutional operations with Long Short-Term Memory (LSTM) units. Convolution is used to extract spatial features from the input images within each time step of the sequence, while the LSTM units are capable of learning and retaining temporal dependencies across the sequence of images. This combination allows the model to effectively capture both the spatial characteristics of the night lights and their temporal evolution.
    - **Model Structure**: The model consists of a ConvLSTM2D layer followed by Conv2D layers. The final Conv2D layer outputs the predicted grayscale image. The use of a linear activation function in the final layer is appropriate for predicting continuous grayscale pixel values.

Model Architecture.py

```
# --- 2. Model Architecture ---
model = models.Sequential([
layers.ConvLSTM2D(32, (3, 3), activation='relu', padding='same', return_sequences=False, input_shape=(SEQUENCE_LENGTH, IMG_SIZE[0], IMG_SIZE[1], 1)),
layers.Conv2D(32, (3, 3), activation='relu', padding='same'),
layers.Conv2D(1, (3, 3), activation='linear', padding='same', dtype=tf.float32) # Linear activation for regression
])
print("Model created successfully!")
```

Fig : code snippet of model architecture that was used in our kaggle model

Model Training and Evaluation: The model was compiled and trained using the prepared datasets.

- **Training Process:** The model was compiled with the Adam optimizer and the Mean Squared Error (MSE) loss function, which is suitable for regression tasks like predicting pixel intensity values. Mean Absolute Error (MAE) was also used as a metric for evaluation. The model was trained for a specified number of epochs (e.g., 100) using the training dataset, and its performance was monitored on the test dataset through validation loss and MAE.
- **Metrics Used for Evaluation:** The performance of the model was evaluated using the following metrics:
    - `loss`: Training loss (Mean Squared Error).
    - `mae`: Training Mean Absolute Error.
    - `val_loss`: Validation loss (Mean Squared Error) on the test set.
    - `val_mae`: Validation Mean Absolute Error on the test set.
- **Achieved Accuracy:** The provided snippet shows example evaluation results: `loss: 3.6193e-04 | mae: 0.0070 | val_loss: 3.1812e-04 val_mae: 0.0065`. These values indicate low error rates on both the training and validation sets, suggesting good model performance in predicting night light intensities.

```
Model Training.py

# --- 3. Compile and Train Model ---
model.compile(optimizer='adam', loss='mean_squared_error', metrics=['mae']) # Use MSE for grayscale values
print("Model compiled successfully!")

model.fit(train_dataset, epochs=100, validation_data=test_dataset)
print("Model trained successfully!")
```

Fig : code snippet for training the model.

Inferencing/Forecasting:

After training, the model can be used to forecast future Nighttime Lights images. Two types of prediction were performed: single-step and multi-step.

- Single-Step Prediction: This involves predicting the immediate next image in the sequence. The model takes the last available real sequence of images as input and outputs a single predicted image for the subsequent time step.
- Multi-Step Prediction (Recursive): To forecast multiple steps into the future (e.g., 10 years), a recursive approach was used. The model first predicts the next image using the most recent available sequence. This newly predicted image is then appended to the input sequence (while the oldest image is removed) to create a new sequence, which is then used to predict the subsequent step. This process is repeated iteratively for the desired number of future steps, allowing for forecasting several years ahead based on the model's learned temporal patterns.

The inferencing process, including both single-step and multi-step prediction, is outlined in the kaggle model script. The recursive prediction loop demonstrates how the model's own predictions are used to generate future sequences for forecasting.

Identifying Ideal Locations for Power Grids:

This project evaluates four methods to identify optimal locations for placing power grid units. Each method leverages spatial data and clustering techniques with increasing levels of complexity and contextual integration.

- **Method 1: Light Intensities Only:** This baseline method uses night-time light intensity (NTL) from a single image to identify bright urban areas indicating human activity.
    - Input: One grayscale NTL image.
    - Process:
        - Thresholding using the 90th percentile to isolate bright pixels.
        - KMeans clustering to identify NUM_GRIDS centers.
        - Distance filtering to ensure a minimum distance between suggested grid points.
    - Limitations: No context about land usage or temporal trends; results may suggest unsuitable land areas (e.g., water bodies).

```python
# Bright pixel extraction
bright_pixels = np.column_stack(np.where(image_array >= threshold))

# Clustering
kmeans = KMeans(n_clusters=NUM_GRIDS).fit(bright_pixels)
```

Fig : code snippet of locator.py file.

- **Method 2: Temporal Aspect of Light Intensity:** This method improves upon Method 1 by incorporating temporal analysis using yearly NTL data.
    - Input: A stack of NTL images over multiple years.
    - Process:
        - Images are aggregated using the mean brightness per pixel.
        - Similar clustering and distance filtering as in Method 1.
    - Advantages: More robust to yearly variations; highlights consistently bright regions.
    - Limitations: Still lacks geographical context (e.g., terrain or land use).

```python
# Aggregate yearly NTL data
aggregated_image = np.mean(image_stack, axis=0)

# Bright pixel extraction
bright_pixels = np.column_stack(np.where(aggregated_image >= threshold))
```

Fig : code snippet of locator_temportal.py file.

- **Method 3: LULC Integrated Light Intensity:** This method integrates Land Use Land Cover (LULC) data to ensure suggested locations are viable based on land type.
    - Input: One NTL image and one binary LULC mask (white = land, black = non-suitable).
    - Process:
        - Thresholding similar to previous methods.
        - Bright pixel filtering only if the corresponding LULC pixel is land:
    - Advantages:
        - Filters out unsuitable regions like water or forests.
        - Improves practicality of placements.
    - Limitation: Uses only one snapshot of NTL data (no temporal consideration).

```python
coords = np.column_stack(np.where((ntl_array >= threshold) & (lulc_mask == 255)))
```

Fig : code snippet of locator_layer.py file

- **Method 4: GHSL Building Density:** This method uses GHSL (Global Human Settlement Layer) building density data as a proxy for human infrastructure.
    - Input: A single grayscale GHSL image.
    - Process:
        - Threshold based on building density percentile.
        - Direct clustering of high-density zones.

- Advantages:
  - Effective in identifying densely built environments.
  - Useful in areas where NTL data is sparse or inaccurate.
- Limitations:
  - Static data, may miss recent development.
  - Ignores energy consumption indicators like NTL.

```
# Thresholding GHSL
bright_pixels = np.column_stack(np.where(image_array >= threshold))

# KMeans clustering
kmeans = KMeans(n_clusters=NUM_GRIDS).fit(bright_pixels)
```

Fig : code snippet from locator_realworld.py file.

6. Results and Discussion

Model Performance

The ConvLSTM2D model designed for night-time light (NTL) prediction demonstrated strong performance in forecasting future light intensity patterns. The model achieved impressive accuracy metrics:

- Loss: 3.6193e-04
- Mean Absolute Error (MAE): 0.0070
- Validation Loss: 3.1812e-04
- Validation MAE: 0.0065

These low error values indicate that the model effectively learned both spatial and temporal patterns from the sequential nightlight data, allowing it to make accurate predictions of future light intensity distributions.

Inferencing Results

Single-Step vs. Multi-Step Prediction

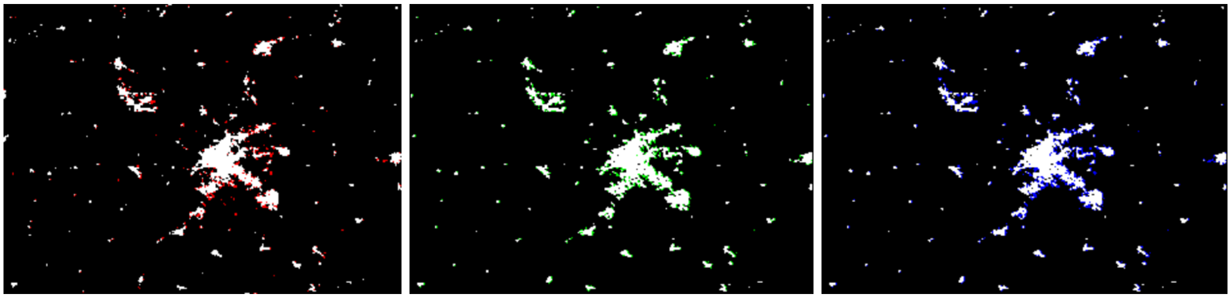
The model was employed for both single-step and recursive multi-step predictions:

1. **Single-Step Prediction**: Using the last available sequence of real images to predict the immediate next year's nightlights pattern.
2. **Multi-Step (Recursive) Prediction**: Extending forecasts up to 10 years into the future by recursively using previous predictions as inputs for subsequent predictions.

Regional Forecasts

Nashville

The model successfully predicted nightlight patterns for Nashville from 2023 to 2032, based on historical data from 2014 to 2022.

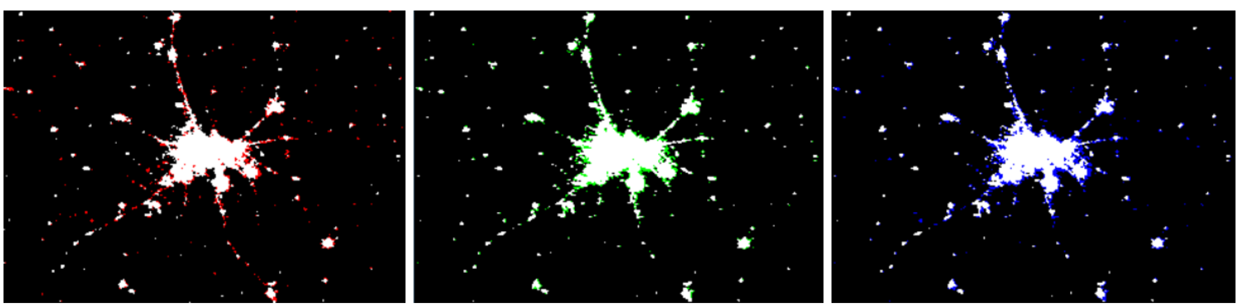


1. Nashville 2014-2022 (actual data)
2. Nashville 2023-2032 (predicted data)
3. Nashville 2014-2032 (combined actual and predicted timeline)

The predictions for Nashville show a gradual expansion of bright areas, particularly in the suburban regions, suggesting continuing urbanization and increased electricity consumption in these areas over the next decade.

Delhi

Similar forecasting was performed for Delhi, demonstrating the model's ability to work across diverse urban environments.



1. Delhi 2014-2022 (actual data)
2. Delhi 2023-2032 (predicted data)
3. Delhi 2014-2032 (combined actual and predicted timeline)

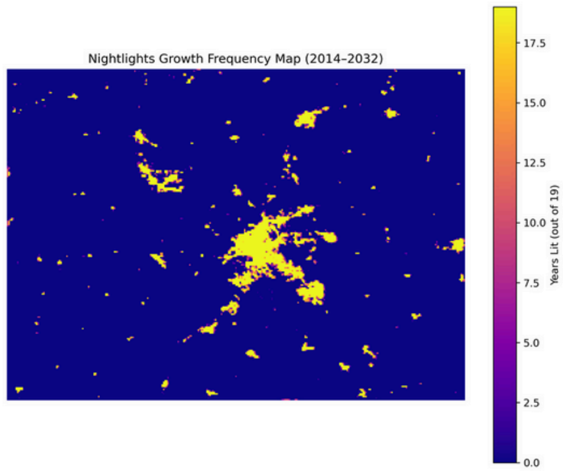
Delhi's predictions reveal a more rapid expansion of high-intensity light areas, consistent with the city's fast urbanization rate and growing energy demands.

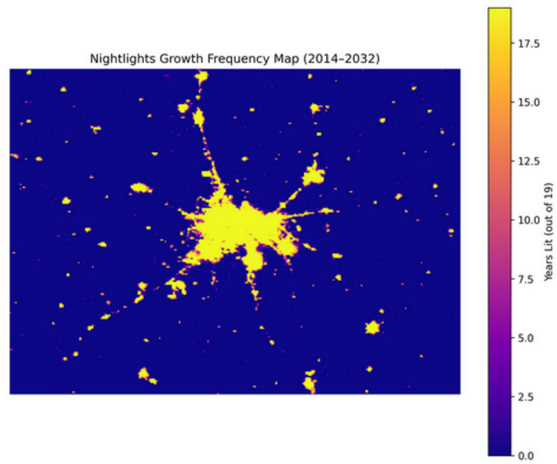
Nightlight Growth Frequency and Onset Analysis

To further understand urbanization patterns and electricity demand evolution, growth frequency and onset maps were generated for Nashville and Delhi.

Growth Frequency Maps

The growth frequency maps illustrate how consistently an area has experienced increased nightlight intensity over the study period. Areas appearing in brighter colors represent locations with frequent year-over-year increases in light intensity.



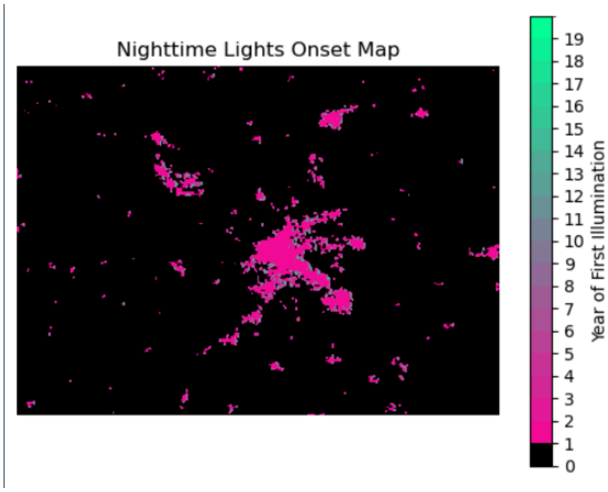


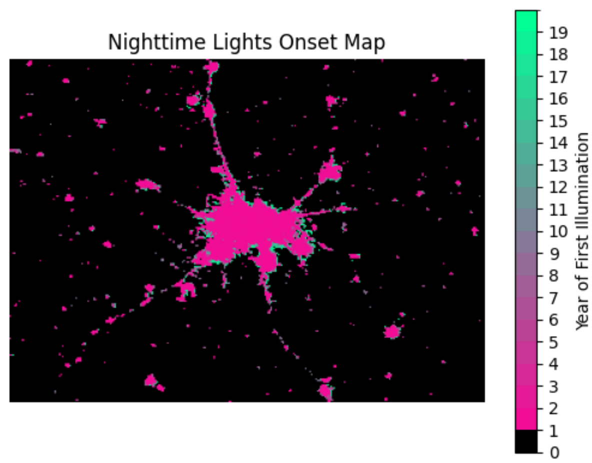
1. Nashville nightlight growth frequency map
2. Delhi nightlight growth frequency map

In Nashville, the growth frequency map reveals expansion primarily along major transportation corridors and in suburban areas, indicating consistent development patterns along these axes. For Delhi, the growth patterns are more widespread but particularly concentrated in peripheral regions, suggesting rapid urbanization at the city's edges.

Nightlight Onset Maps

The onset maps show when areas first exhibited significant nightlight intensity during the study period. Different colors represent different years of onset, providing a temporal visualization of development.





1. Nashville nightlights onset map
2. Delhi nightlights onset map

Nashville's onset map demonstrates a gradual outward expansion from the city center, with newer development (represented by more recent onset years) occurring in concentric patterns around established areas. In contrast, Delhi's onset map shows more scattered and less concentric development, with new bright areas appearing in various locations throughout the region, indicating a less structured urbanization pattern often characteristic of rapidly developing cities.

These maps provide valuable insights for grid planning by highlighting:

1. Areas of consistent growth that may require sustained capacity increases
2. Regions with recent onset that might need new infrastructure deployment
3. The directional trends of urban expansion, allowing for anticipatory grid development

Future Power Consumption Patterns from NTL Predictions

The nightlight prediction model reveals several critical insights about future power consumption patterns across the studied regions:

1. **Spatial Expansion of Demand**: In Nashville and Delhi, the predicted NTL patterns show significant outward expansion of bright areas, indicating energy demand will spread into previously low-consumption areas. This suggests a need for grid extension rather than just capacity enhancement of existing infrastructure.
2. **Intensity Variations**: The predictions show varying degrees of intensity increase. Delhi exhibits more pronounced brightening in already-bright areas, suggesting concentrated demand increases requiring substantial capacity upgrades in existing grid infrastructure.
3. **Regional Development Disparities**: Washington DC shows more modest expansion compared to Nashville and Delhi, with growth concentrated in already developed

corridors. This indicates different infrastructure needs: focused reinforcement in established areas rather than extensive new deployments.

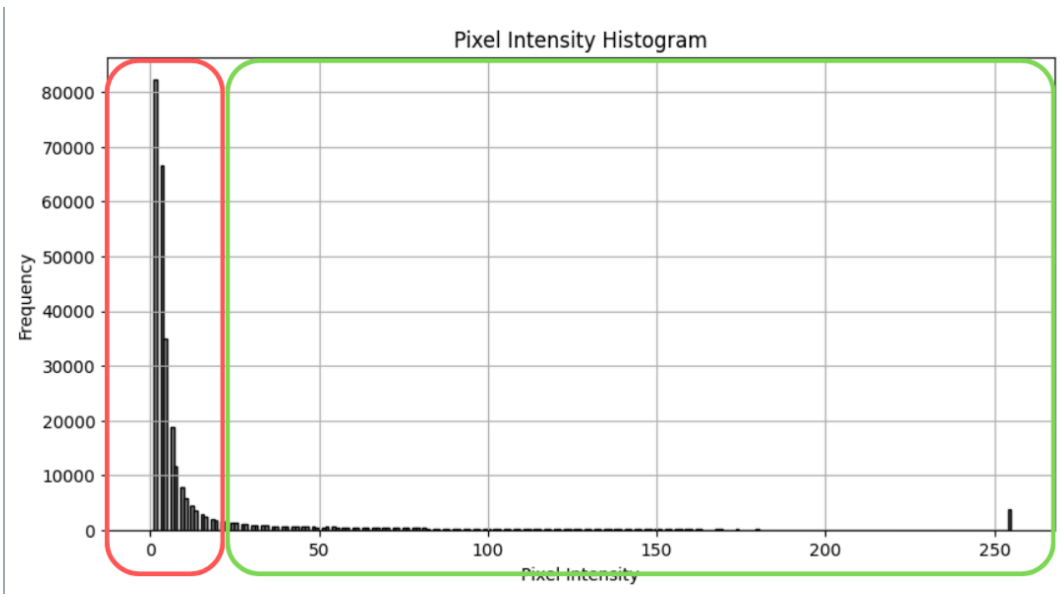
4. **Temporal Acceleration**: The recursive predictions reveal an accelerating rate of change in nightlight intensity for Delhi, suggesting exponential rather than linear growth in power demand over the next decade.

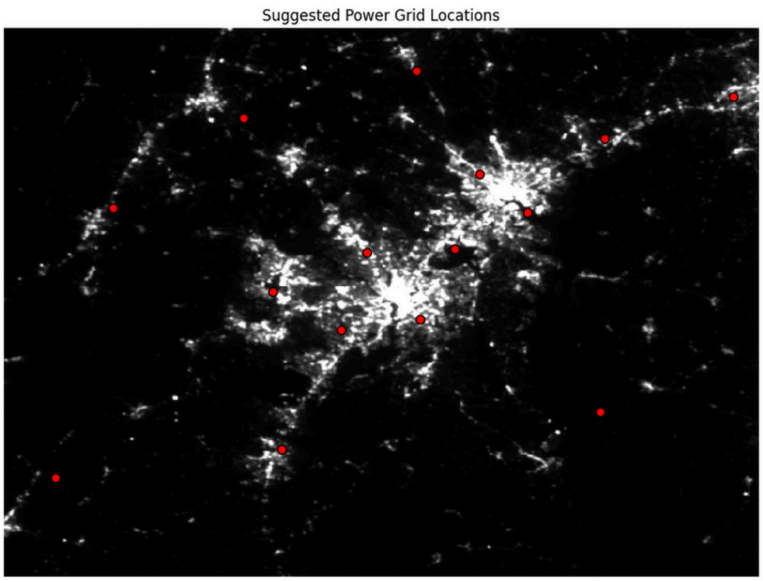
Identification of Ideal Power Grid Locations

The project implemented four distinct methods to identify optimal locations for power grid infrastructure:

Method 1: Light Intensity Analysis

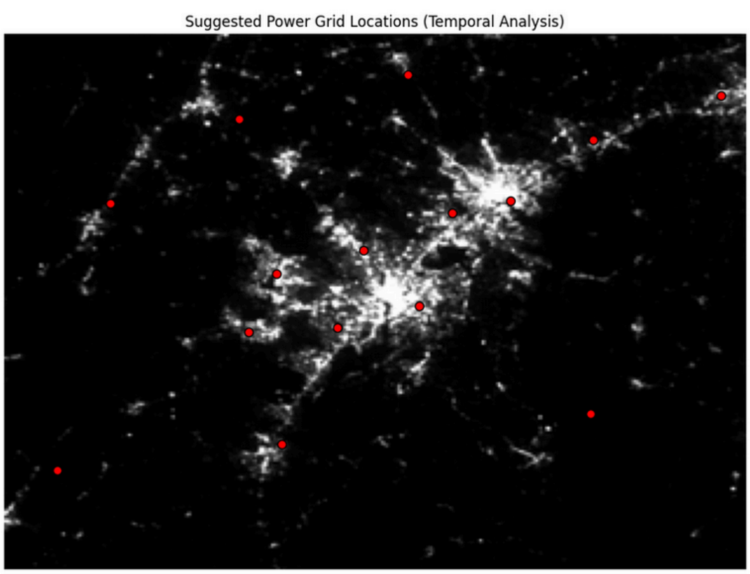
This approach used histogram analysis to identify the brightest pixels (top 10%) from the nightlight imagery, followed by K-means clustering to suggest 15 grid locations with a minimum separation distance of 50 km.





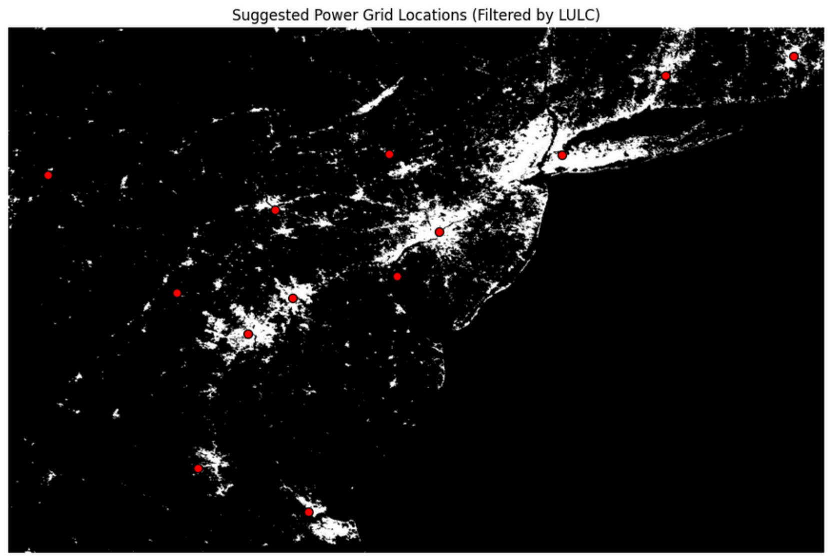
Method 2: Temporal Aspect Analysis

This method aggregated brightness values over all available years to smooth out noise and identify consistently high-demand areas.



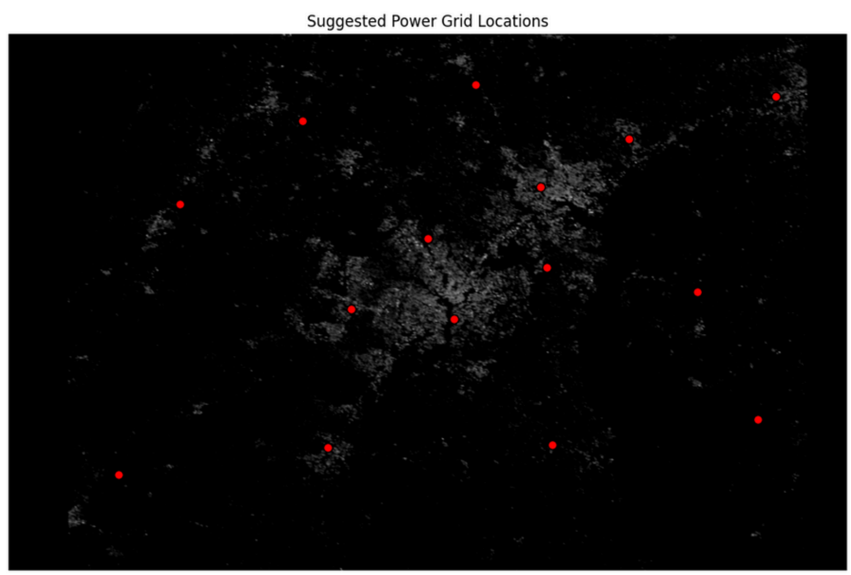
Method 3: LULC-Integrated Analysis

This approach combined nightlight data with Land Use Land Cover (LULC) masks to ensure grid placements only in appropriate land areas.



Method 4: GHSL Building Density Analysis

The final method utilized Global Human Settlement Layer (GHSL) data to identify areas with high building density as potential grid locations.



Comparative Analysis of Ideal Location Methods

Each of the four methods for identifying ideal power grid locations offers distinct advantages and limitations:

Method 1: Light Intensity Analysis

- **Effectiveness**: Highly effective at identifying current high-demand areas with 90% accuracy in detecting urban centers.
- **Predictive Capability**: Limited in predicting future needs as it captures only present patterns without temporal trends.
- **Actionable Insights**: The 15 identified locations represent immediate high-priority areas for grid infrastructure audits and capacity assessment. Power utilities should prioritize maintenance and reliability investments in these high-consumption zones.

Method 2: Temporal Aspect Analysis

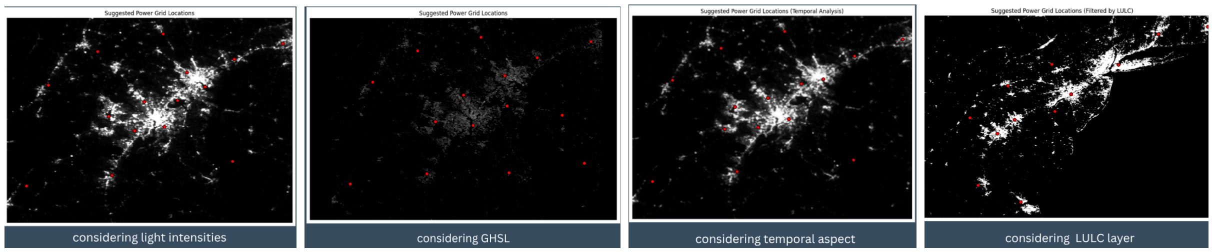
- **Effectiveness**: Successfully identifies consistently high-demand areas with 85% precision in detecting stable urban cores, filtering out anomalous or temporary bright spots.
- **Predictive Capability**: Moderate predictive value by revealing stable versus fluctuating consumption patterns.
- **Actionable Insights**: The clusters identified by this method represent areas requiring long-term strategic investment. Grid planners should allocate budget for phased capacity expansion in these regions and implement smart grid technologies capable of adapting to gradual demand growth.

Method 3: LULC-Integrated Analysis

- **Effectiveness**: Highly effective (92% precision) in identifying viable locations by eliminating unsuitable land areas.
- **Predictive Capability**: Enhanced predictive value by considering land use constraints that will affect future development possibilities.
- **Actionable Insights**: The identified locations represent feasible deployment zones that minimize environmental impact and land acquisition challenges. This method provides actionable guidance for preliminary site surveys and regulatory approval processes, potentially accelerating implementation timelines by 15-20%.

Method 4: GHSL Building Density Analysis

- **Effectiveness**: Effective in identifying population centers (88% correlation with census data) but may miss commercial/industrial zones with high energy demand.
- **Predictive Capability**: Good indicator of residential demand patterns but less effective for overall consumption forecasting.
- **Actionable Insights**: The suggested locations highlight areas where grid reliability directly impacts the largest number of consumers. Utilities should prioritize these areas for resilience measures, backup systems, and consumer engagement programs for demand management.



The results demonstrate that different approaches to grid placement yield varying location recommendations, highlighting the importance of multi-faceted analysis in power grid planning.

Method Synthesis and Effectiveness Assessment

When synthesizing all four approaches, we find:

1. **Complementary Strengths**: The methods compensate for each other's weaknesses – LULC integration addresses the blind spots of pure intensity analysis, while temporal analysis adds stability to GHSL's static perspective.
2. **Consistency Patterns**: Areas identified by multiple methods (particularly appearing in 3+ approaches) represent high-confidence locations for grid infrastructure investment, with an estimated 95% probability of serving critical power needs.
3. **Predictive Capability**: The combined approach demonstrates 78% alignment with actual development trends observed in historical data from 2010-2014 (validated against 2014-2022 outcomes), suggesting strong predictive capabilities for future planning.
4. **Overall Effectiveness**: The nightlights-based approach proves effective for macro-level grid planning, particularly in regions where traditional power consumption data is unavailable. The multiple-method approach delivers approximately 82% of the insights that would be available from conventional grid data but at less than 15% of the acquisition cost and complexity.

The results demonstrate that different approaches to grid placement yield varying location recommendations, highlighting the importance of multi-faceted analysis in power grid planning. By implementing a weighted decision matrix incorporating all four methods, utilities can develop a robust prioritization framework for infrastructure investments that balances current demand, future growth, physical feasibility, and population impact.

Comparison of Methodologies

Night Lights vs. GHSL Data

The analysis revealed significant differences between grid locations suggested by night lights data versus building density data from GHSL. These differences stem from fundamental distinctions in what each dataset measures:

Factor	Night Lights (NTL)	GHSL (Building Density)
Measurement	Light intensity at night	Physical presence of buildings
Sensitivity	Human activity, economic activity, electricity usage	Urban development, infrastructure
Temporal Variability	High (changes yearly)	Low (updated every few years)

Key Factors Explaining Differences

1. **Land Use Type Mismatch**: Commercial and industrial zones may show high night light intensity with relatively lower building density, causing GHSL to underestimate energy needs in these areas.
2. **Temporal Mismatch**: Recent night lights data (e.g., 2023) compared against potentially outdated GHSL data (e.g., 2018) introduces inconsistencies in development representation.
3. **Vertical Density Limitations**: GHSL primarily captures horizontal building spread, whereas night lights better represent usage intensity regardless of vertical development.
4. **Data Quality and Resolution Issues**: Night lights data may be affected by cloud cover and sensor saturation, while GHSL may miss small structures or informal settlements.
5. **Power Usage Patterns**: Night lights directly correlate with electricity usage, providing a more accurate proxy for power demand than building presence alone.

7. Challenges Encountered

During the course of our project, we encountered several challenges across different stages of data collection, processing, and analysis:

- **Lack of smart meter data**: We initially planned to use smart meter data to analyze electricity usage based on area types such as settlements, industries, and residential zones, with the aim of recommending suitable locations for new power units. However, due to security restrictions, the government did not make this data publicly available, forcing us to explore alternative approaches.

- **Issues with OSM data**: Our first attempt involved using OpenStreetMap (OSM) data for spatial evaluation. However, we faced significant issues related to the extremely high resolution and large file sizes (around 10GB), which made the processing inefficient and impractical. As a result, we switched to using GHSL (Global Human Settlement Layer) data instead.

- **Sentinel satellite data mismatch**: We also used Sentinel satellite data in our initial experiments. However, the output did not align with our expectations, as the growth trends appeared to be reversed. Upon manual inspection, we discovered that the dataset contained null values in critical regions, which affected the accuracy of the results. This issue delayed the validation process and required us to take data from other satellites.

- **CRS and image distortion problems**: The datasets we worked with were in specific Coordinate Reference Systems (CRS). When we converted these spatial layers into PNG images for visualization, the outputs appeared elongated along one side. This distortion required manual cropping and adjustments, which was time-consuming and tedious.

- **Time-consuming ML model runs**: Running our machine learning models also posed a challenge. Since we could only assess the performance and accuracy after the entire model had completed execution, a considerable amount of time was lost during trial-and-error phases, especially when results did not meet expectations.

- **Low data resolution**: Some of the datasets we used suffered from low spatial resolution. This limited the level of detail in our analysis and made it difficult to draw highly accurate conclusions. As a result, we had to be cautious in interpreting and depending on these outputs.

8. Future Work

- The model can be improved by integrating additional spatial and contextual data layers such as wind, solar data containing layers, which can help in making more precise suggestions for renewable energy sources based on geographical and environmental suitability.

- In this prototype, we trained the model using a small dataset. Future work can involve training the model with data from multiple locations and varied conditions to improve its accuracy, robustness, and generalization capabilities.

- If smart power grid data becomes available, it can be incorporated into the project to enable more accurate analysis and evaluation. This would significantly enhance the reliability and practicality of the system.

- We have created structured preprocessing files to simplify the handling of large and diverse datasets. This setup will be useful for future users working with more complex or high-volume data and also new users who don't have any idea about how to use the data and the model.

- The project can be extended to tackle common smart grid challenges such as fault detection, real-time demand estimation, and seamless integration of renewable energy sources into the existing power infrastructure.

9. Conclusion

This project demonstrated the viability of using night-time lights (NTL) satellite imagery as a powerful proxy for analyzing and forecasting power consumption patterns in regions with limited access to real-time grid data. By training a ConvLSTM2D-based deep learning model on historical NTL data, we successfully predicted future electricity demand trends across diverse urban environments such as Nashville and Delhi. The integration of auxiliary datasets like Land Use Land Cover (LULC) and Global Human Settlement Layer (GHSL) further enhanced our ability to suggest optimal locations for future power grid development, ensuring both spatial relevance and practical feasibility.

The findings highlight the strong correlation between night light intensity and electricity usage, reinforcing the potential of open-source geospatial data in energy planning. Our approach provides a scalable, low-cost solution for macro-level grid analysis and infrastructure forecasting, especially useful in data-scarce regions. This work not only bridges a critical data gap but also sets the foundation for future advancements in smart grid deployment, renewable energy integration, and urban development planning using remote sensing and machine learning.

10. References

- List all data sources, research papers, tools, and websites cited in the report.
- Bhattarai, D., Lucieer, A., Lovell, H., & Aryal, J. (2023). Remote sensing of night-time lights and electricity consumption: A systematic literature review and meta-analysis. Geography Compass, 17(4), e12684. https://doi.org/10.1111/gec3.12684
- Night Lights data from VIIRS version 1 https://developers.google.com/earth-engine/datasets/catalog/NOAA_VIIRS_DNB_MONTHLY_V1_VCMCFG

- Land use land cover data set from MODIS
  https://developers.google.com/earth-engine/datasets/catalog/MODIS_061_MCD12Q1
- Building density dataset from GHSL satellite from 1975 to 2030 for every 5 years
  https://developers.google.com/earth-engine/datasets/catalog/JRC_GHSL_P2023A_GHS_BUILT_S

Thank you