**Crime Hotspot Prediction**

**Project Idea :-**

Our project is particularly aimed at providing a more protected society by predicting crime based on past history of data. Digitization of all information will ultimately provide our working model with enough data to predict crime accurately. Increasing task force in given areas based on Machine learning will be the true amalgam of technology and security.

**Our Aim – Predict future hotspots and alert different crime agencies.**

**Project Requirements :-**

* For marking crime hotspots we need to identify the area to be considered and the maximum jurisdiction area that can be covered by a police department to further increase the security by deploying more forces and stricter checking and patrolling.
* After getting such data we can map the crime hotspots to each of these area and then identify the area to predict the hotspots for the next month based on these data provided.
* Now the next thing required will be to implement this using a proper mapping model like a GIS software. Sending the data through a machine learning pipeline to predict the hotspots will be possible based on the success and the time taken by the grid creation and mapping techniques.
* Creating a proper machine learning model using the aggregated data

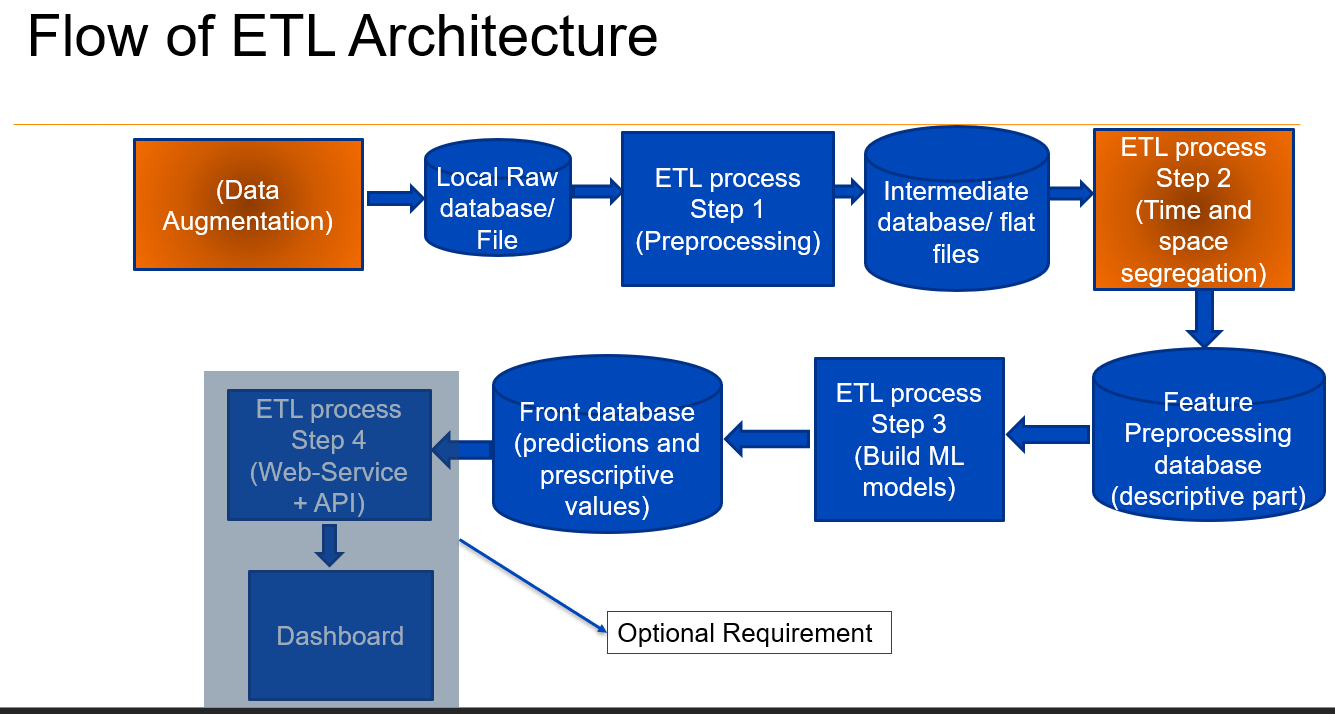
**Innovation –**

Such a method of adopting a grid based technique for predicting crimes and applying a machine learning model to that has never been planned before. It has been analysed that approaching this way will give us better accuracy and proper crime zone prediction.

**Proposed Address Augmentation Technique (If required):-**

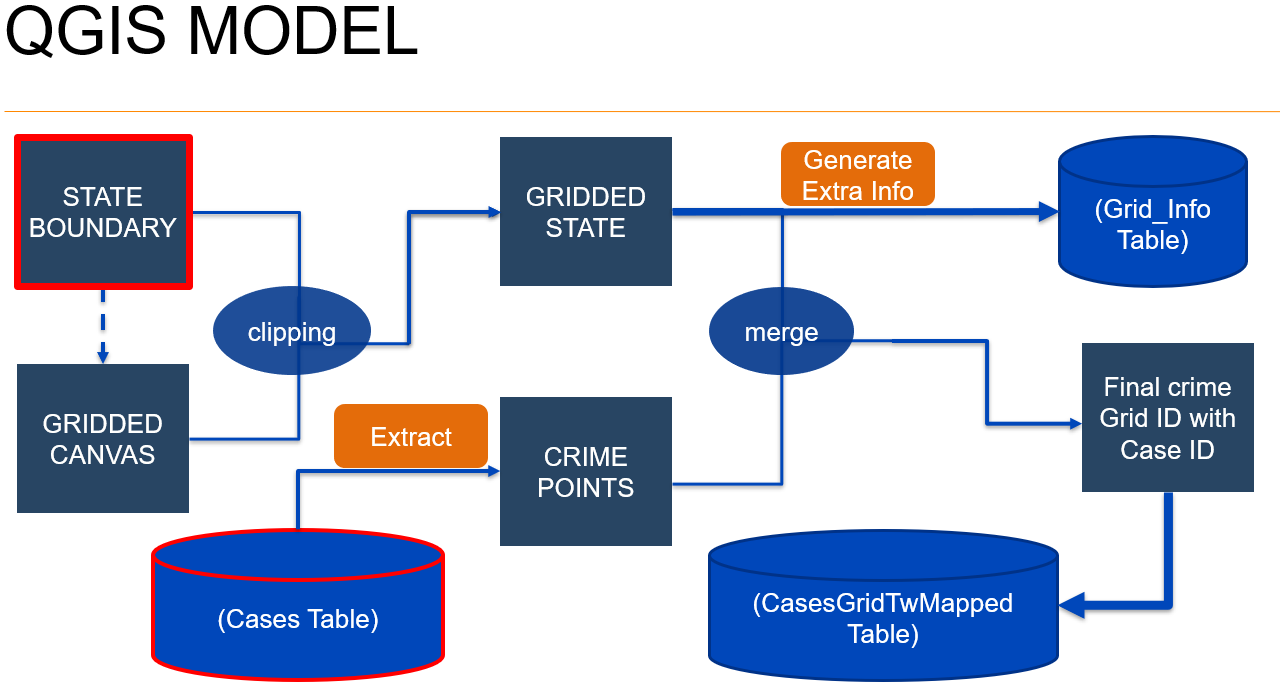
The Address Augmentation module deals with extracting data from the Raw database and to the local database and referring every address to the locally stored cache or if not present, then firing the Nominatim API to get proper latitude and longitude. Each such address is looked up first in the cache containing distinct addresses and if not available only then is the API fired. On getting multiple lat – lon pairs the pairs are averaged based on a regular heuristic. Otherwise they are set to (0, 0) which indicates that it was either not found, or the distance between points are too large. This is how incremental data of addresses are handled.

**Entire Extract Transform Load(ETL) architecture of our project:-**



* **Step** **1** – After possible address augmentation we add the incremental monthly crime data as and when supplied by the agency and add it to the existing flat file for the next monthly prediction.
* **Step** **2** – Now after getting all the crime data along with the adjoining (lat,lon) pairs we assign a month id from the beginning of the data timestamp for every month. Also a particular grid id is assigned to every crime point so that a particular area can be predicted as a future hotspot
* **Step** **3** – After assembling grid\_id , month\_id, and crime case id in a single table we use this merged data and feed it for training and testing using a suitable machine learning model for the required prediction

**The model shown is what we aim to achieve as a initial mapping model by replicating the Quantum Geographic Information System (QGIS) into our python codes.**



The Final output from this module will be the entire crime points mapped to each and every grid which will be necessary for the design of our machine learning model in the future.

**Need of modified QGIS using python:**

* Because the entire software of QGIS includes various different vector layer functions which are not necessary for our project
* Also, Using QGIS toolkit every time for mapping out a county/state is not a feasible solution since we need a single pipeline to work out the entire model

**Importance:**

The QGIS model’s efficiency will decide the overall time taken by the entire planned pipeline and ensures the working of each component

(The entire model shown above will be a part of feature preprocessing module in the architecture mentioned previously)

**Problems Faced:**

* Initial Grid Boundary could not be formed due to GDAL (Geospatial Data Abstraction Library) environment error which was later resolved by using proper Conda environment
* Initial Pipeline creation did not take QGIS into consideration as a different module
* Figuring out a method to create the initial clipped out grid similar to QGIS
* Mapping model merging not working due to improper projection system

**Contribution of Different Project Members (till now) –**

Ranajay Saha: - Project Planning and Documentation

(1651172)

Niladri Purkait: - Architecture Pipeline Modelling

(1651187)

Siddhant Chakraborty: - QGIS Model workout and modularization

(1651173)

Shreyasis Roy: - Project Design,

(1651176) QGIS Implementation and Modularization, and

Pipeline Modelling