Data Preparation/Feature Engineering

Project Title: Offline AI-Powered Community Support and Vulnerability Prediction System for Conflict-Affected Areas in Myanmar

Team Members (Actively Participated):

1. **Thant Zin Moe**
2. **Aye Yu Yu San**

1. Overview

This document presents the data preparation and feature engineering phase of the project, being the most critical part of the machine learning pipeline, as the quality and relevance of the data directly determine the performance of the final vulnerability prediction model. This section will cover the collection of raw data, the steps taken to clean and process it, an exploratory data analysis (EDA) to find patterns, and the feature engineering methods used to create predictive inputs for our models.

2. Data Collection

This project is built upon a diversity of data and the datasets collected are:

Quantitative Dataset (For the Prediction Model)

i. **ACELED Data** - A CSV file ("`ACLED Data_2025-11-06.csv`") containing raw conflict event data for Myanmar from 2018 to 2025, including event types, locations and fatalities.
ii. **World Bank Data** - Two CSV files: one is metadata ("`3399bf22-a9a4-45b1-8fdc-3aedadafa775_Series - Metadata.csv`") and the actual data ("`3399bf22-a9a4-45b1-8fdc-3aedadafa775_Data.csv`") are collected containing national-level indicators for Myanmar, such as GNI per capita and poverty metrics.
iii. **UNDP Data** - A CSV file ("`undp_hdi.csv`") is collected which include Human Development Index data for Myanmar.
iv. **Satellite Data** - Nighttime Lights (from the Payne Institute) and Precipitation (from CHIRPS) for the 2018-2024 period is collected as folders containing GeoTIFF with ("`.tif`") files.

Qualitative Dataset (For the Chatbot)

This dataset is a custom-curated knowledge base, mutually collected by a team member who reviewed and extracted relevant FAQs and guidance on health, education and aid from authoritative NGO and UN websites, representing as data specifically collected for the chatbot component.

3. Data Cleaning

This phase is essential to transforming the raw data into a structured, reliable format for modeling and analysis, involving several key steps:

- **Loading and Inspection** – Each of the four datasets (ACLED, World Bank, UNDP, and the custom Chatbot data) is loaded into a pandas Data frame for inspection.
- **Column Selection** – The columns relevant to the project will be selected to reduce complexity. For example, the ACLED data has over 30 columns, but we only need about 8-10, such as "event_date", "region", "admin1", "latitude", "longitude", and "fatalities".
- **Handling Missing Values** – The missing data will be checked with a strategy that is to drop any rows from the ACLED data which are missing critical location information, since they cannot be mapped.
- **Data Type Conversion** – The chosen columns will be converted to their proper data types, such as changing "event_data" column in the ACLED file from a simple string to a "datetime" object, which is essential for time-series analysis and so on and so forth.
- **Standardization and Reshaping** – The world bank data will be "unpivoted" from a wide format (years as a column) to a long format (on row per indicator per year) making it usable. And, for the chatbot's custom data is standardized to a consistent format such as "Health", "Education".
- **Exploring Clean Data** – This is the final state, in which the cleaned Data Frames will be saved to the "`Data/processed/`" folder, separating them from the raw data.

4. Exploratory Data Analysis (EDA)

The first thing here is the Time-Series Insights which is shown at the following figure showing a massive spike in conflict after 2021.

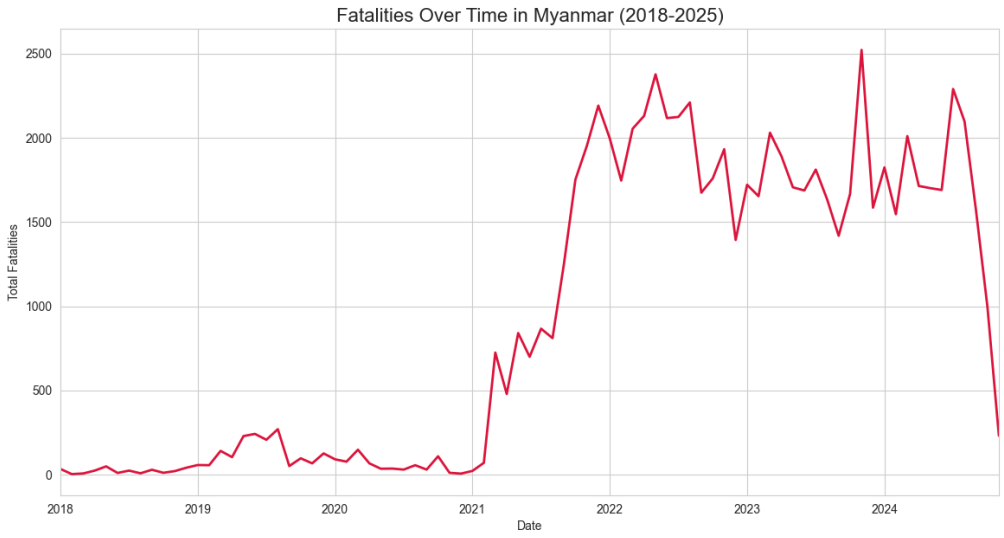


Fig 1: Fatalities Over Time in Myanmar

The following figure shows the event insights showing which events such as “Battles” or “Violence” are the most common.

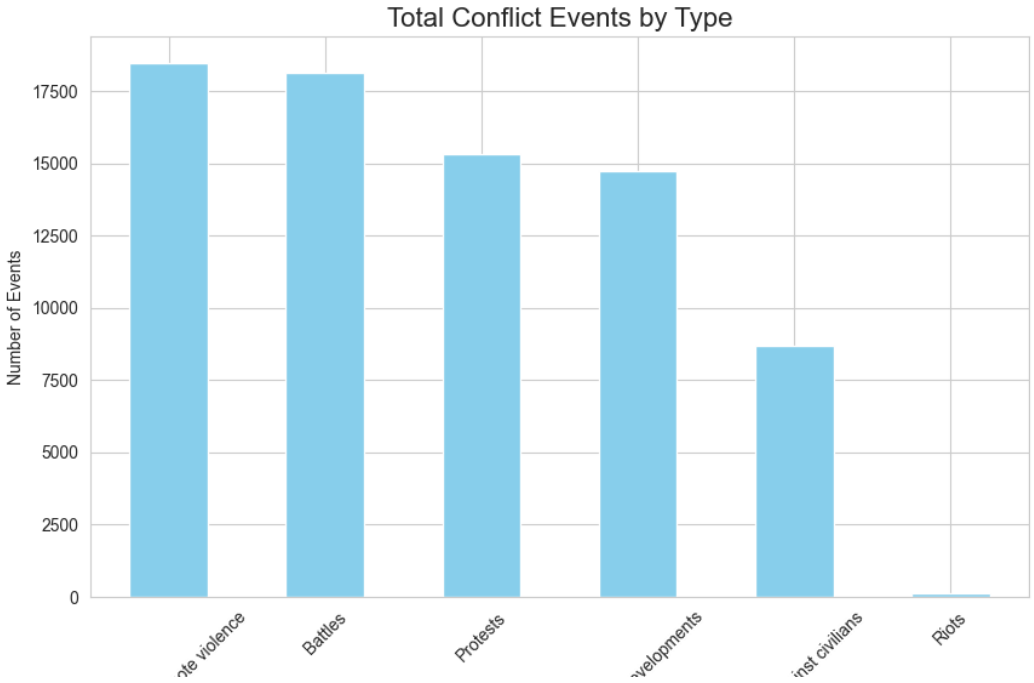


Fig 2: Conflict Events Sorted and Visualized by Type

The final part is the Geographic insights as demonstrated below showing that vulnerability is not evenly spread.

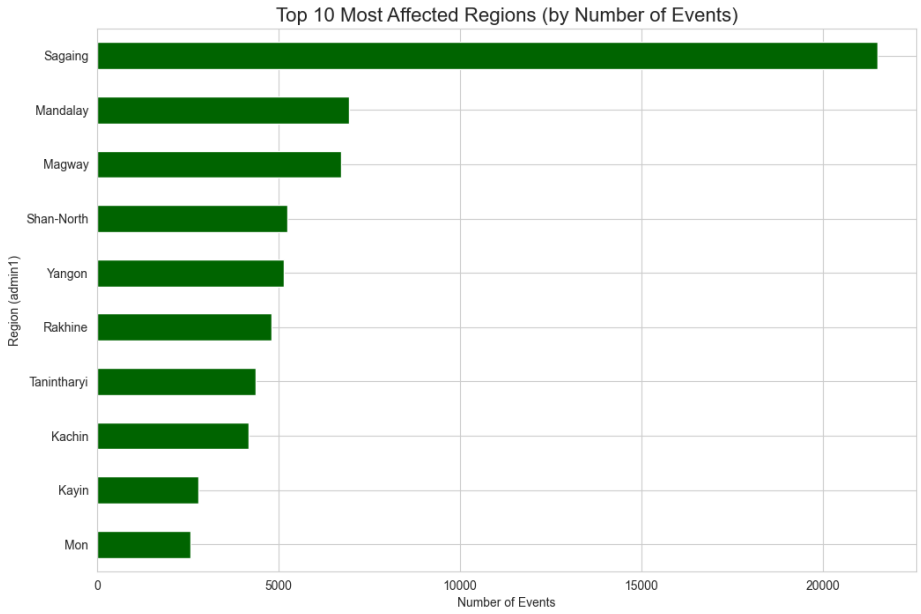


Fig 3: Top Affected Region of Myanmar

5. Feature Engineering

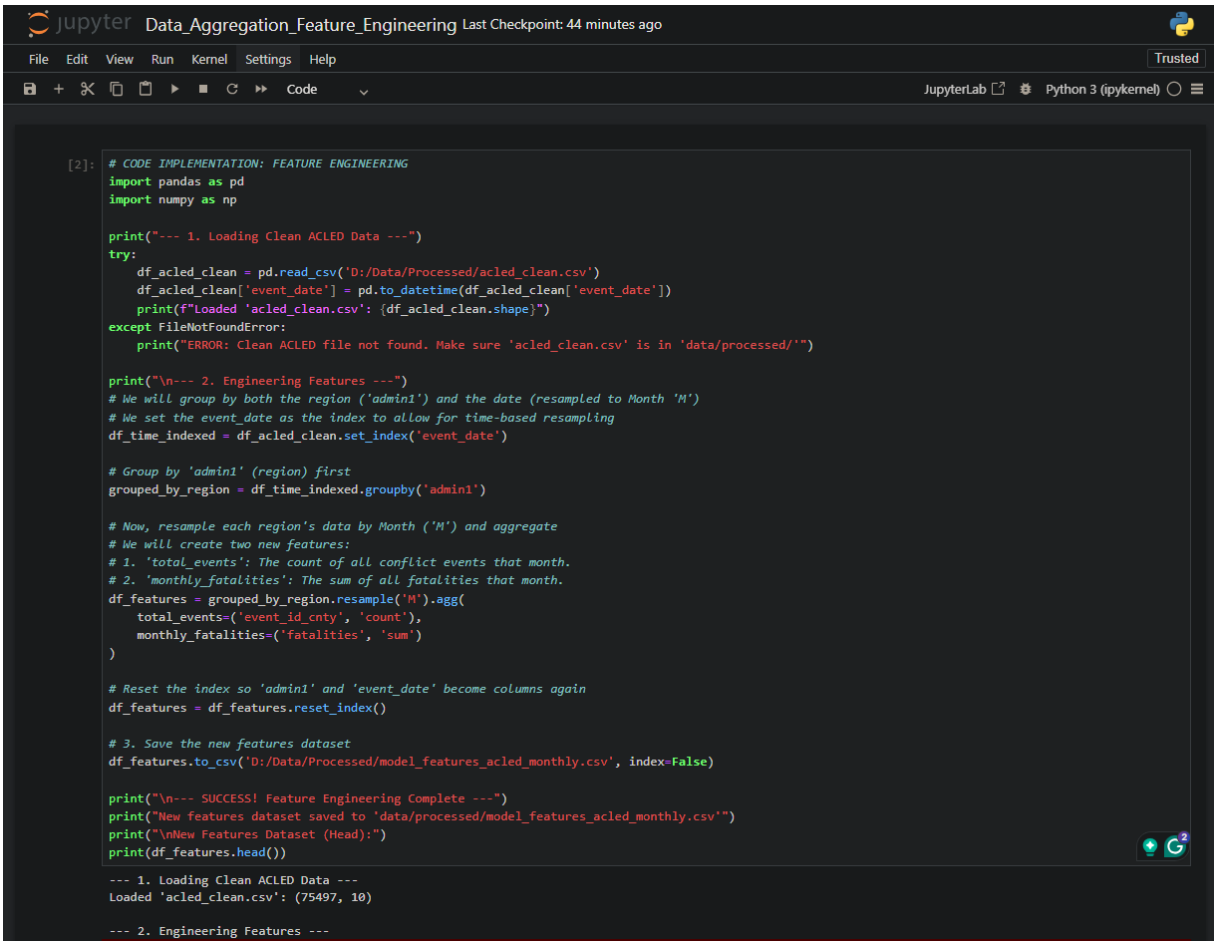
The raw event data is not in a suitable format for a machine learning model, requiring a fixed set of features for each geographic unit. The main goal of this phase is to engineer new features by aggregating the event-level data from the "`aceled_clean.csv`" file to a regional and monthly level, involving a step of processes

Temporal Aggregation – This is about converting daily event data into a consistent monthly time-series using "**resample()**".

Geospatial Aggregation – In this, individual events are grouped by their "admin1" region such as "Shan" or "Sagaing".

Feature Creation – This is about creating new and predictive features from this aggregation such as "monthly_fatalities" and "total_conflict_events".

The code snippet is as shown:

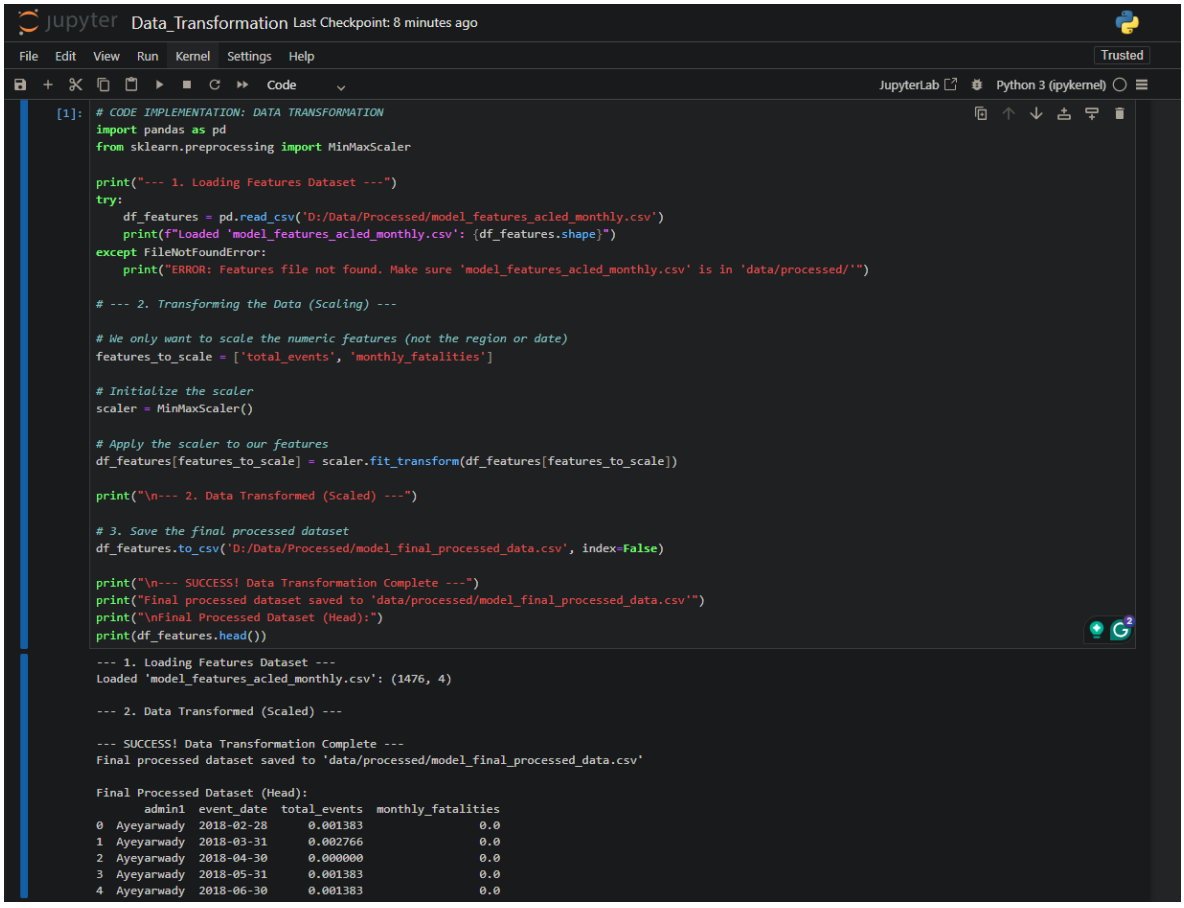


6. Data Transformation

Data transformation, specifically normalization or standardization, is a required step for many machine learning models, especially deep learning models like 1D-CNN. The new features (like total_events and monthly_fatalities) have very different scales, which can confuse the model.

"**Min-Max Scaling**" will be used for this project, transforming all feature values to be within a specific range (e.g., 0 to 1). This ensures that all features contribute equally to the model's learning process and improves model stability and training speed.

The code snippet for this process is shown below.



Model Exploration

1. Model Selection

As described, two different machine learning models will be explored and compared in this project's methodology.

Model 1: Random Forest Regressor

This model was chosen as a powerful and interpretable baseline.

- **Strength**: Random Forest is excellent at handling tabular data, is robust to outliers and can provide feature importance scores, which will help in understanding that factors are the most significant predictors of vulnerability.
- **Weakness**: This model is not natively designed for understanding temporal sequences.

Model 2: 1D-Convulutional Neural Network

This model was chosen specifically because the data of our project is a time-series and feature engineering created a dataset of monthly conflict metrics over time.

- **Strength**: As included in the literature review, 1D-CNNs are highly effective at efficiently extracting temporal correlations from sequential data, learning patterns like small spike in “protests” or large spike in “fatalities” that the Random Forest might miss.
- **Weakness**: This model is harder to interpret, kind of a “Blackbox”, and more complex to set up and train.

Both models will be trained using the same data, comparing the performance of each to see which model or a hybrid of both is best suited for this project.

2. Model Training

The models will be trained to predict “monthly_fatalities” using the other features (like “total_events” and “region”).

Data Splitting: Since the data is Time-Series, a random train_test_split cannot be used, as this would cause data leakage (using future data to predict the past). And thus, Temporal Split will be used.

- Training Set: All data from 2018-01-01 to 2023-12-31.
- Testing Set: All data from 2024-01-01 to the present.

This method simulates a real-world scenario where we use the past to predict the future.

Cross-Validation: For hyperparameter tuning, **TimeSeriesSplit** from **scikit-learn** will be used, which is the correct cross-validation technique for temporal data.

3. Model Evaluation

Assessment of the models' performance is done using the following standard regression metrics:

- **Root Mean Squared Error (RMSE)**: This is the primary metric which measures the average magnitude of the errors, giving a higher weight to large (and more dangerous) prediction errors.
- **Mean Absolute Error (MAE)**: This measures the average error in the same units as the target (e.g., "the model is, on average, off by 10 fatalities").

- **Visualization**: The most important evaluation will be a plot of Actual vs. Predicted fatalities on the test set. This will visually show how well our model's predictions track with reality.

4. Code Implementation

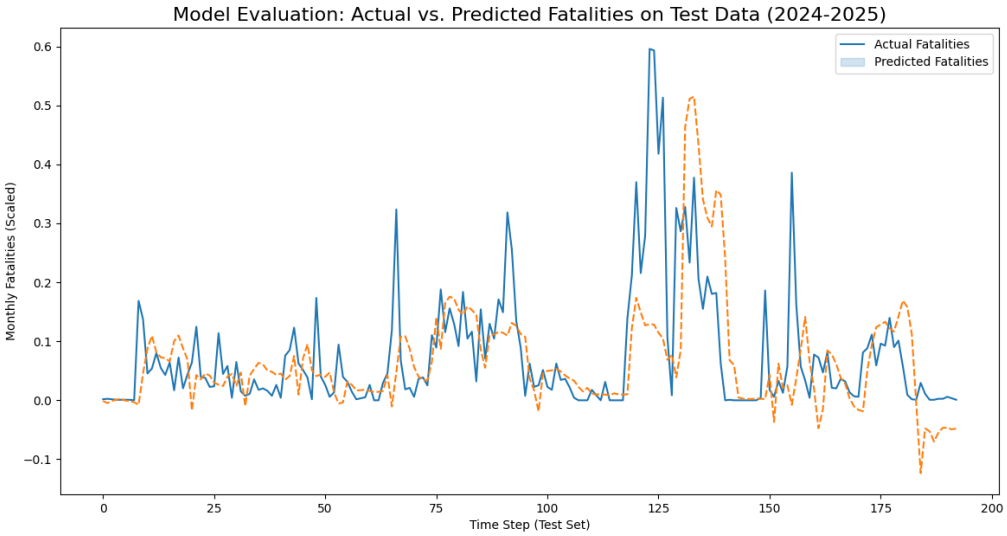


Fig 4: Model Evaluation Visualization of Random Forest vs 1D-CNN

This is for the baseline model of this project and later, we need to enrich or enhance the model with different data we have collected so far.

For the code snippets I will attach a link which will redirect you to all the files for this assignment. The reason why I am not including the data files in this assignment is because the data files I collected are too large.

Code Snippets –

[https://drive.google.com/drive/folders/1OyVhQubgJIB2Dy0FzfmzVpYYWNHt589m?usp=sharing]