Project Update on Earthquake Building Damage Prediction System

Presented By:
Anish Shilpakar
Anushil Timsina
Aarosh Dahal
Sugam Karki

Overview

- ❑ Work Completed
- ❑ Work in Progress
- ❑ Work Remaining

Work Completed

- ❏ Dataset Collection
- ❏ Data Cleaning
- ❏ EDA
- ❏ Data preprocessing

Dataset Collection

- Multiple datasets were collected from Kaggle.com
  Link: Predicting Building Damage from Earthquakes | Kaggle
- Currently, working on csv_building_structure.csv dataset
- The dataset has 31 columns with both numeric and categorical variables.
- Target Column: damage_grade (5 grades: Grade 1, Grade 2, Grade 3, Grade 4, Grade 5)

Dataset Information

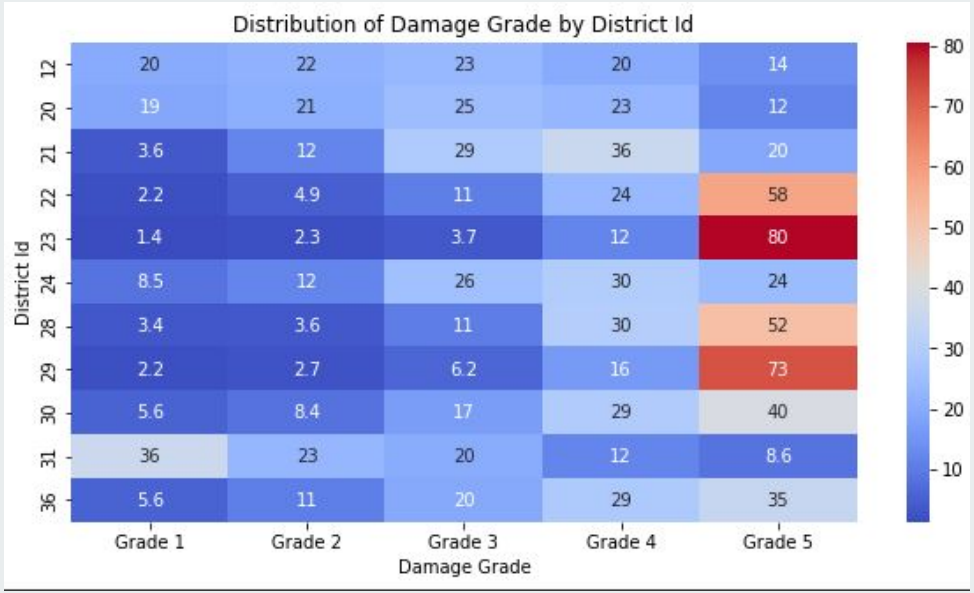
```
Data columns (total 31 columns):
 #   Column                                  Non-Null Count   Dtype
---  ------                                  --------------   -----
 0   building_id                             762106 non-null  int64
 1   district_id                             762106 non-null  int64
 2   vdcmun_id                               762106 non-null  int64
 3   ward_id                                 762106 non-null  int64
 4   count_floors_pre_eq                     762106 non-null  int64
 5   count_floors_post_eq                    762106 non-null  int64
 6   age_building                            762106 non-null  int64
 7   plinth_area_sq_ft                       762106 non-null  int64
 8   height_ft_pre_eq                        762106 non-null  int64
 9   height_ft_post_eq                       762106 non-null  int64
 10  land_surface_condition                  762106 non-null  object
 11  foundation_type                         762106 non-null  object
 12  roof_type                               762106 non-null  object
 13  ground_floor_type                       762106 non-null  object
 14  other_floor_type                        762106 non-null  object
 15  position                                762105 non-null  object
 16  plan_configuration                      762105 non-null  object
 17  has_superstructure_adobe_mud            762106 non-null  int64
 18  has_superstructure_mud_mortar_stone     762106 non-null  int64
 19  has_superstructure_stone_flag           762106 non-null  int64
 20  has_superstructure_cement_mortar_stone  762106 non-null  int64
 21  has_superstructure_mud_mortar_brick     762106 non-null  int64
 22  has_superstructure_cement_mortar_brick  762106 non-null  int64
 23  has_superstructure_timber               762106 non-null  int64
 24  has_superstructure_bamboo               762106 non-null  int64
 25  has_superstructure_rc_non_engineered    762106 non-null  int64
 26  has_superstructure_rc_engineered        762106 non-null  int64
 27  has_superstructure_other                762106 non-null  int64
 28  condition_post_eq                       762106 non-null  object
 29  damage_grade                            762094 non-null  object
 30  technical_solution_proposed             762094 non-null  object
dtypes: int64(21), object(10)
```

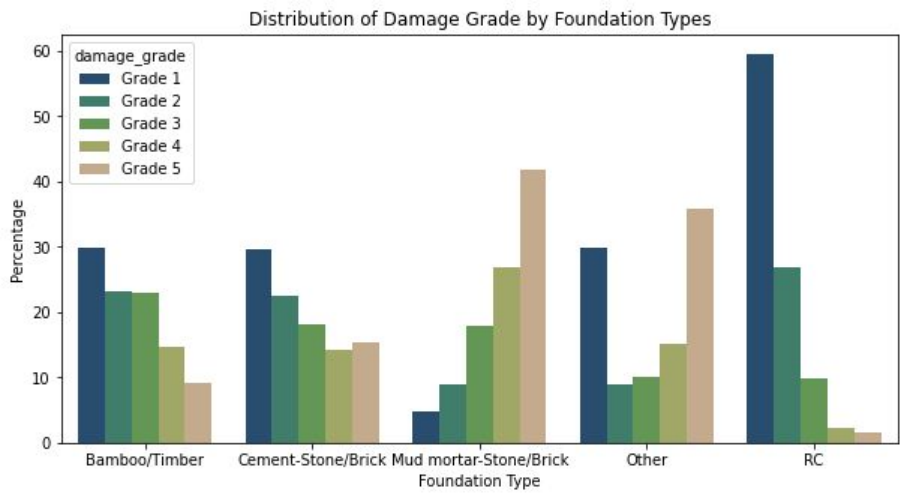
Data Cleaning

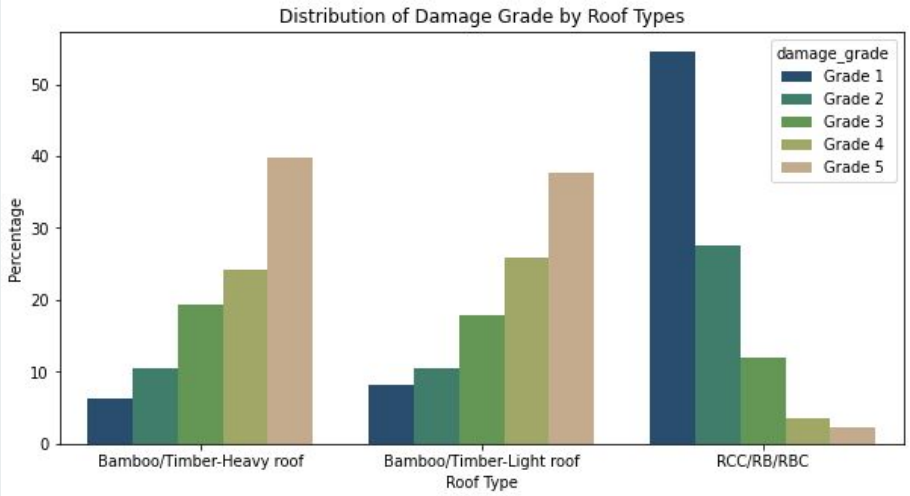
- In this step, the dataset was analyzed for null values.
- Null values were found on the categorical attributes, so they were replaced by mode.
- Also, some null values were found on the target column and these rows were dropped.
- These reduced overall rows to 762094.

EDA

- ❏ Basics of EDA
- ❏ Distributions explored
- ❏ Further EDA will be done.




Distribution of Damage Grade by Foundation Types
damage_grade
Grade 1
Grade 2
Grade 3
Grade 4
Grade 5
Percentage
0
10
20
30
40
50
60
Bamboo/Timber
Cement-Stone/Brick
Mud mortar-Stone/Brick
Other
RC
Foundation Type


Distribution of Damage Grade by Roof Types
damage_grade
Grade 1
Grade 2
Grade 3
Grade 4
Grade 5
Percentage
0
10
20
30
40
50
Bamboo/Timber-Heavy roof
Bamboo/Timber-Light roof
RCC/RB/RBC
Roof Type

Data Preprocessing

- In this step, variable encoding was done for categorical variables
- Both nominal and ordinal data is present in dataset.
- Label encoding the ordinal attributes using Scikit Learn’s LabelEncoder.
- One hot encoding the nominal attributes using Pandas get_dummies() function.

Work in Progress

- ❏ Feature Selection
- ❏ Model Creation
- ❏ Model Evaluation

Feature Selection

- Feature Selection using correlation
  - Evaluated using all present features
  - Evaluated using top 5 features
- Feature Scaling using Standard Scaler to bring all attributes to similar range and to reduce computational cost.

Model Creation

- Considered 3 machine learning models from sklearn as of now:
  - Logistic Regression
  - Decision Tree
  - Random Forest

Model Evaluation

- Considered parameters like Precision, recall and f1 score.
  - Logistic Regression

Considering all features

	precision	recall	f1-score	support
0	0.70	0.73	0.71	19713
1	0.47	0.17	0.24	21958
2	0.55	0.72	0.63	34066
3	0.77	0.85	0.81	45959
4	0.97	0.95	0.96	68831
accuracy			0.77	190527
macro avg	0.69	0.68	0.67	190527
weighted avg	0.76	0.77	0.76	190527

Considering top 5 features

	precision	recall	f1-score	support
0	0.61	0.65	0.63	19828
1	0.24	0.06	0.09	21828
2	0.54	0.70	0.61	33963
3	0.74	0.87	0.80	46028
4	1.00	0.95	0.97	68877
accuracy			0.75	190524
macro avg	0.63	0.65	0.62	190524
weighted avg	0.73	0.75	0.73	190524

Model Evaluation

- Decision Tree

Considering all features

	precision	recall	f1-score	support
0	0.92	0.92	0.92	19713
1	0.78	0.78	0.78	21958
2	0.75	0.75	0.75	34066
3	0.84	0.84	0.84	45959
4	0.97	0.97	0.97	68831
accuracy			0.87	190527
macro avg	0.85	0.85	0.85	190527
weighted avg	0.87	0.87	0.87	190527

Considering top 5 features

	precision	recall	f1-score	support
0	0.97	0.66	0.79	19828
1	0.67	0.85	0.75	21828
2	0.79	0.74	0.76	33963
3	0.80	0.91	0.85	46028
4	1.00	0.95	0.97	68877
accuracy			0.86	190524
macro avg	0.85	0.82	0.82	190524
weighted avg	0.87	0.86	0.86	190524

Model Evaluation

- Random Forest

Considering all features

	precision	recall	f1-score	support
0	0.96	0.91	0.94	19713
1	0.83	0.85	0.84	21958
2	0.81	0.82	0.81	34066
3	0.86	0.90	0.88	45959
4	0.99	0.96	0.98	68831
accuracy			0.90	190527
macro avg	0.89	0.89	0.89	190527
weighted avg	0.91	0.90	0.91	190527

Considering top 5 features

	precision	recall	f1-score	support
0	0.98	0.66	0.79	19828
1	0.67	0.85	0.75	21828
2	0.79	0.74	0.76	33963
3	0.80	0.91	0.85	46028
4	1.00	0.95	0.97	68877
accuracy			0.86	190524
macro avg	0.85	0.82	0.82	190524
weighted avg	0.87	0.86	0.86	190524

Model Evaluation

- Confusion Matrix

Work Remaining

- More EDA
- Model Optimization
  - Hyperparameter Tuning
  - Selection of Best Model
- Model Deployment
- Documentation

Thank You!!!