IJCRT.ORG ISSN : 2320-2882

INTERNATIONAL JOURNAL OF CREATIVE RESEARCH THOUGHTS (IJCRT)

An International Open Access, Peer-reviewed, Refereed Journal

Machine Learning-Based Earthquake Damage Assessment For Buildings: A Study With Catboost, Adaboost, Xgboost And Lightgbm

Rahi Bhand, Rushikesh Ghodke, Tanmay Bora, Viraj Zuluk and Manisha Mali
Vishwakarma Institute of Information Technology, Pune, India

Abstract— In the face of unpredictable natural calamities, the capacity to quantify structural damage with precision could redefine the boundaries of post-disaster recovery and resource allocation. Automating the classification of building damage can expedite the process of distributing aid and reduce reliance on manual inspection. Accurate damage assessment is critical not only for post-earthquake recovery but also for taking pre-emptive measures to strengthen vulnerable structures. In this paper, we explore machine learning techniques such as AdaBoost, CatBoost, and a stacking ensemble to classify earthquake-induced building damage using data from the Gorkha earthquake. The methodology incorporates steps such as feature encoding, model training, and ensemble learning to enhance prediction accuracy. Key structural features—including building age, number of floors, foundation type, area percentage, and secondary usage—are integrated into the models to assess the predicted damage severity.

Index Terms— Earthquake Damage Prediction, Building Damage Assessment, Ensemble Learning, CatBoost, XGBoost, AdaBoost, LightGBM, Machine Learning, Stacking Ensemble, Structural Attributes, Gorkha Earthquake Dataset

I. INTRODUCTION

In recent years, earthquakes have become more frequent and violent, leading to significant infrastructure disruptions, financial losses, and fatalities worldwide. Some prominent examples are the 2010 Haitian earthquake, the 2015 Nepalese Gorkha earthquake, and the more recent 2023 earthquakes in Syria and Turkey, which all resulted in significant destruction of buildings and casualties [1][2]. Evaluation of structural damage must be done quickly and accurately in order to prepare for recovery, respond to disasters, and rebuild. Traditional assessment methods, while useful, sometimes rely on subjective and limited expert judgement and factual observations.

In order to improve forecast accuracy, this study presents an ensemble technique that combines LightGBM with CatBoost and AdaBoost, two machine learning models for assessing building damage after an earthquake. This work intends to fill in the gaps left by earlier approaches and provide a dependable and scalable solution for practical applications in post-disaster damage assessment by emphasizing efficient feature processing, including categorical variables, and enhancing model interpretability.

This study presents an ensemble technique that combines CatBoost and AdaBoost with LightGBM to improve prediction accuracy, as well as a machine learning-based approach to evaluating building damage after an earthquake. This work intends to fill in the gaps left by earlier approaches and provide a dependable and scalable solution for practical applications in post-disaster damage assessment by emphasizing efficient feature processing, including categorical variables, and enhancing model interpretability.

II. LITERATURE SURVEY

The previous work focused on developing machine learning models to predict the level of building destruction following the 2015 Gorkha earthquake in Nepal. The researchers employed a range of classification methods, including logistic regression, k-nearest neighbours (k-NN), random forest, and XGBoost[2][1], to estimate damage grades based on attributes including building age, construction type, and geographic location.

The XGBoost model outperformed the other algorithms under study, accurately forecasting damage levels. The research highlights the significance of feature selection and parameter modification in enhancing model performance.

The performance of the models is significantly influenced by the completeness and quality of the dataset. If the dataset lacks diversity or is biassed, the model's predictions may be incorrect [2].

The prior study examines a number of machine learning techniques for evaluating earthquake damage, stressing the efficacy of various approaches in forecasting damage levels. It talks about how forecast accuracy may be improved by using ensemble techniques, such as integrating many models. The study mostly addresses theoretical methods without offering empirical support for their applicability or validity in the actual world.

In order to improve the predictions' generalisability, our model will make use of a more extensive and varied dataset that spans several earthquake occurrences and geographic areas. Our strategy will make use of sophisticated ensemble approaches that strike a compromise between interpretability and accuracy, enabling a deeper comprehension of the model's predictions while preserving strong performance.

III. DATASET OVERVIEW

The **Gorkha Earthquake in Nepal** provided the dataset utilized in this study, which offers comprehensive details on a range of building attributes affected by the quake. Its main purpose is to forecast the buildings' damage assessment score, or damage_grade, which is divided into three groups:

- **Grade 1 Damage**: Minor harm
- **Grade 2 Damage**: Damage that is moderate
- **Grade 3 Damage**: Severe damage

Numerous characteristics pertaining to building construction, location, and other elements that could affect the extent of damage after an earthquake are included in the dataset.

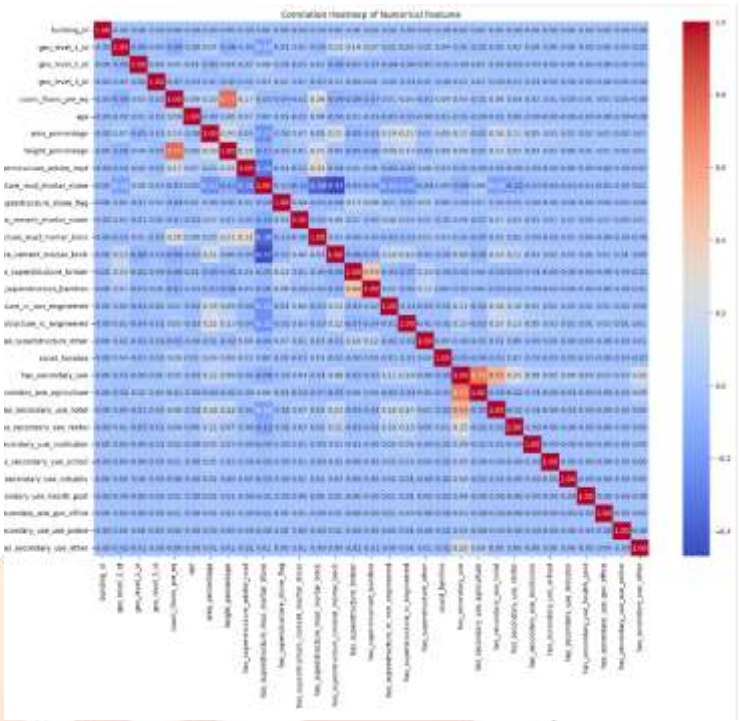

Fig 1. Correlation Heatmap of Numerical

A. *Key Features:*

The dataset's features may be roughly divided into:

- **Building characteristics:** Things like height, age, type of roof, kind of foundation, etc.
- **Geographic Details:** Geographical level, district, and position are examples of location-based qualities.
- **Structural and Non-structural characteristics:** Whether the building has specific structural reinforcements or secondary functions, such as commercial or agricultural.

B. *Feature Descriptions and Their Importance*

A key component of figuring out which factors have the most effects on the damage assessment score is feature importance. We determined which characteristics were most pertinent by using LightGBM for feature significance analysis. An overview of the salient features and their relative significance may be seen below.

Attribute	Description	Feature Importance (Sample)
geo_level_1_id	Geographic level 1 identification of the building location	High (Top 3)
geo_level_2_id	Geographic level 2 identification, more granular than level 1	High (Top 5)
geo_level_3_id	Geographic level 3 identification, the most specific location	Moderate
age	Age of the building (years)	High
area_percentage	Percentage of the total area covered by the building	High
height_percentage	Height of the building relative to other structures	High
foundation_type	Type of foundation used in the building construction	Moderate
roof_type	Type of roof used (e.g., bamboo, metal, etc.)	Moderate
ground_floor_type	Material used for the ground floor (e.g., cement, mud, etc.)	Low
other_floor_type	Material used for floors other than the ground floor	Low
position	Position of the building (e.g., attached, detached)	Moderate
plan_configuration	Shape/configuration of the building layout	Moderate
has_superstructure_mud_mortar_stone	Whether the building has a superstructure made of mud mortar stone	Low
has_superstructure_cement_mortar_brick	Whether the building has a superstructure made of cement mortar bricks	High
has_secondary_use	Whether the building has secondary uses like agriculture or business	Low
has_secondary_use_agriculture	Whether the building is used for agricultural purposes	Low

C. Feature Importance Analysis

Each attribute's decrease in impure (Gini significance)[10] was utilized to determine the feature priority using LightGBM. The most significant predictors of the damage grade were found to be the building's age and geo_level_1_id [2], indicating that a structure's age and location significantly affect its vulnerability to earthquake damage. Factors like area_percentage and height_percentage also received good scores since larger, taller buildings usually incur more severe damage.

Interestingly, despite their importance, structural attributes like roof_type and foundation_type had a relatively little impact [1]. According to the has_superstructure_cement_mortar_brick feature, which possesses a major influence, buildings with cement mortar brick constructions are more resilient to earthquake damage.

Feature significance is a crucial element in determining which aspects have the greatest impact on the damage assessment score. We used LightGBM for feature significance analysis to identify the most relevant attributes. Below is a summary of the key characteristics and their relative importance.

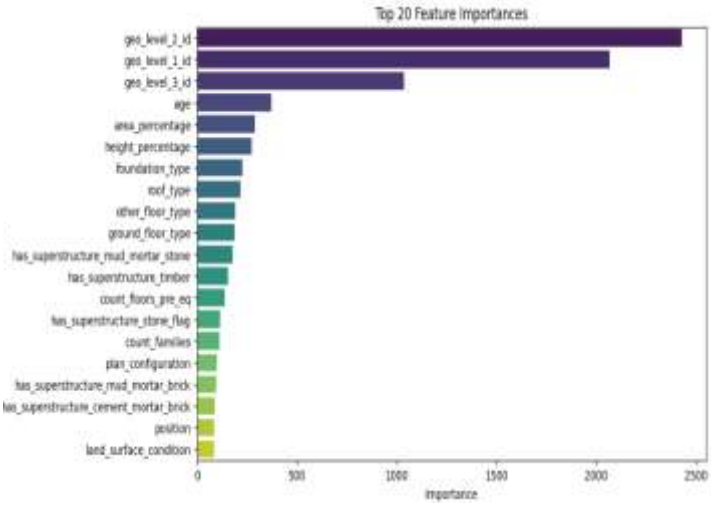

Fig 2. Feature Importance

IV. METHODOLOGY

This study forecasts the damage grades of structures using a powerful machine learning approach based on a range of structural and geographic parameters. Among the key steps that comprise the approach are data preprocessing, feature engineering, model selection, hyperparameter adjustment, and performance evaluation.

A. Data Acquisition and Preprocessing

The training and testing values and corresponding labels that comprise the datasets utilized in this study were loaded using the Pandas library. Prior domain expertise and exploratory data analysis were used to choose the pertinent characteristics [7], and the datasets were retrieved in a structured CSV format. The prediction's target variable is the damage grade of the buildings.

B. Feature Selection

For model training, a subset of features was chosen, with an emphasis on variables including building attributes, structural components, and geographic identifiers. Among the chosen features are:

- Geographical ids: geo_level_1_id, geo_level_2_id, geo_level_3_id
- Building attributes: count_floors_pre_eq, age, area_percentage, and height_percentage
- Structural types: land_surface_condition, foundation_type, roof_type, etc.

C. Target Encoding and One-Hot Encoding

Target Encoding was used well to handle geographic ids and other high-cardinality data categories [7]. By using this technique, the model may learn from the relationship between the category variables and the target variable. Using One-Hot Encoding, additional category attributes were also converted into a suitable number format for model training.

D. Feature Engineering

Feature engineering was performed to generate interaction terms and discarded features that might enhance model performance. Notably, interaction features such as age_floors_interaction and area_height_ratio were created [2]. The height_percentage and age variables were further discarded into categorical groupings (young, middle-aged, elderly, short, medium, and tall) to capture non-linear relationships.

E. Feature Scaling

To guarantee that each feature contributes equally to model performance, numerical features were standardized using the StandardScaler. For algorithms that are sensitive to the size of the input data, this phase is essential.

F. Handling Class Imbalance

The dataset showed class imbalance due to the characteristics of the damage grading [6]. In order to solve this problem and guarantee a balanced dataset for training, synthetic samples for the minority classes were created using the Synthetic Minority Over-sampling Technique (SMOTE)[12][1].

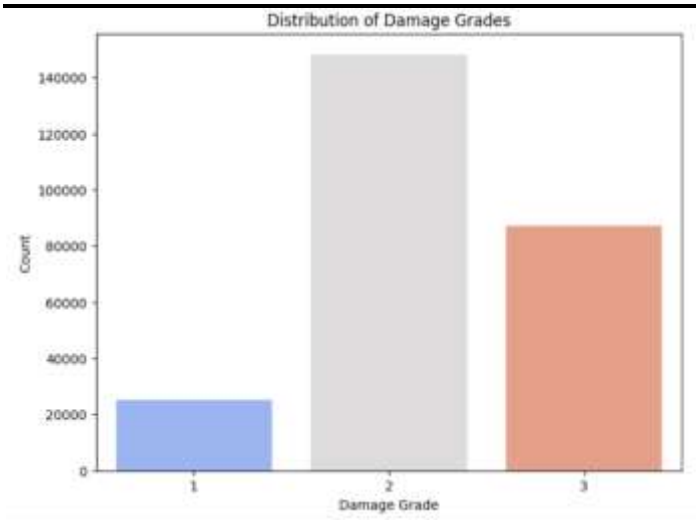

Fig 3. Distribution of Damage Grades

G. *Model Selection and Hyperparameter Tuning*

Multiple machine learning models were explored for this task, including:

- **CatBoost**: A gradient boosting algorithm that handles categorical features natively[3].
- **AdaBoost**: An ensemble method that combines multiple weak classifiers.
- **XGBoost**: An optimized gradient boosting algorithm designed for speed and performance.
- **LightGBM**: A gradient boosting framework that uses tree-based learning algorithms and is optimized for efficiency[5].

Hyperparameter tuning for the CatBoost model was performed using GridSearchCV, optimizing parameters such as iterations, learning rate, and tree depth to enhance model accuracy.

H. *Stacking and Voting Classifier*

Utilizing a Random forest model as the meta-learner, a Stacking Classifier was constructed to capitalize on the capabilities of individual models. CatBoost, AdaBoost, XGBoost, and LightGBM were among the base models. In order to increase overall accuracy, a Voting Classifier [9] was also used for soft voting, combining the estimates from all base models.

I. *Model Evaluation*

A hold-out validation set was used to assess the models and gauge each classifier's accuracy. Predictions generated on the validation results were used to produce accuracy ratings, and the model that performed the best was chosen for further examination. The approach guarantees a thorough assessment of the ensemble models' predictive power for estimating earthquake damage [6].

V. MATHEMATICAL FORMULATIONS

A. *CatBoost Classifier*

CatBoost (Category Boosting) is a gradient boosting technique that performs particularly well with categorical data. Using the idea of decision trees, it improves accuracy and allows the model to deal with categorical features natively by transforming categorical parameters into numerical formats during training.[3] [8].

Foundation of Mathematics:

CatBoost minimizes the following objective function for each iteration *t*:

$$L^{(t)} = \sum_{i=1}^{n} l\left(y_i, F^{(t-1)}(x_i) + \eta h_t(x_i)\right) + \Omega(h_t)$$

- $L^{(t)}L^{(t)}$ is the loss at the current iteration t,
- ll is the loss function (usually log loss for classification),
- $F^{(t-1)}(x_i)F^{(t-1)}(x_i)$ is the prediction from the previous iteration,
- η is the learning rate,
- $h_t h_t$ is the newly learned weak learner,
- $\Omega(h_t)\Omega(h_t)$ represents regularization.

B. *AdaBoost Classifier*

The boosting approach AdaBoost (Adaptive Boosting) transforms weak learners, usually decision trees, into strong learners by repeatedly improving the model's accuracy [12]. It assigns weights to incorrectly classified samples to emphasize their importance in the next iteration.
Mathematical Foundation:

AdaBoost minimizes the exponential loss function:

$$L = \sum_{i=1}^{n} \exp(-y_i F(x_i))$$

- LL is the total loss,

- $y_i\, y_i \in \{-1,1\}$ is the true label of instance i,
- $F(x_i)\, F(x_i)$ is the predicted label from the ensemble.

At each step, AdaBoost updates the weights of misclassified instances to focus more on hard-to-classify examples.

Update rule:

$$w_i^{(t+1)} = w_i^{(t)} \cdot \exp\left(\alpha_t \cdot I\left(y_i \neq h_t(x_i)\right)\right)$$

where:

- $\alpha_t\, \alpha_t$ is the weight assigned to the weak learner,
- $II\,(\cdot)$ is the indicator function, which is 1 if the condition is true and 0 otherwise.

C. XGBoost Classifier

Regularization, tree cutting, and GPU acceleration are all included in XGBoost (Extreme Gradient Boosting), an improved gradient boosting technique [11]. It is highly renowned for its speed and performance on large datasets.
Foundation of Mathematics:

XGBoost minimizes the following regularized objective function:

$$L = \sum_{i=1}^{n} l\left(y_i, \widehat{y_i^{(t)}}\right) + \sum_{k=1}^{T} \Omega(f_k)$$

- $l\, l$ is the loss function (e.g., logistic loss for binary classification),
- $y_i^{(t)}\, y_i^{(t)}$ is the predicted output at iteration t,
- $\Omega(f_k) = \gamma T + 1/2\lambda \sum_j w_j^2 \Omega(f_k) = \gamma T + 1/2\lambda \sum_j w_j^2$ is the regularization term with $\gamma\, \gamma$ controlling the number of leaf nodes TT, and λ controlling the leaf weights.

D. LightGBM Classifier

Histogram-based decision tree training is used in the gradient boosting framework LightGBM.[5]. Due to its support for parallel processing and GPU acceleration, it is built for great performance and efficiency, especially when working with huge datasets.
Mathematical Foundation:

LightGBM employs a similar boosting structure, but to effectively handle huge data, it makes use of Exclusive Feature Bundling (EFB) and Gradient-Based One-Side Sampling (GOSS)[10]. LightGBM's goal function may be expressed as follows:

$$L = \sum_{i=1}^{n} l(y_i, \hat{y}_i) + \sum_{k=1}^{T} \Omega(f_k)$$

Where:

- ll is the loss function (log loss or squared error for classification),
- $\Omega(f_k)\Omega(f_k)$ is the regularization term,
- TT is the number of trees,
- $f_k f_k$ is the k-th decision tree.

E. Stacking Classifier

Stacking is an ensemble learning method that combines the predictions of multiple base classifiers using another model, often called the "meta-learner"[6]. In this case, RandomForestClassifier is used as the meta-learner.
Mathematical Foundation:

Stacking works as a two-layer model. The first layer consists of base learners, and their predictions are passed as features to the meta-learner, which makes the final prediction.

Given base learners $h_1, h_2, \ldots, h_n h_1, h_2, \ldots, h_n$, the meta-learner is trained on:

$$z = \sum_{i=1}^{n} \alpha_i h_i(x)$$

Where:

- zz is the output feature vector from base learners,
- $\alpha_i \alpha_i$ is the weight for the ii-th base learner.

F. Voting Classifier

Another ensemble method that combines the predictions of several classifiers is voting [9]. Both hard voting (majority voting) and softer voting (probability averaging) are supported. Particularly with classifiers that provide probabilistic predictions, such as CatBoost and XGBoost, soft voting is typically more reliable.

\

Mathematical Foundation (Soft Voting):

Soft voting averages the predicted class probabilities:

$$P(y = k) = \frac{1}{n}\sum_{i=1}^{n} P_i(y = k)$$

Where $P_i(y = k)P_i(y = k)$ is the predicted probability of class kk from the ii-th model, and nn is the number of models.

VI. RESULT

The ensemble learning approach, which consists of CatBoost, AdaBoost, XGBoost, and LightGBM, was evaluated for its capacity to predict the degree of building damage caused by earthquakes. The training and assessment dataset consisted of the structural features of the Gorkha earthquake.

The models were trained using layered ensemble approaches and sophisticated feature encoding techniques. Each model worked well by itself, and when coupled, they further improved forecast accuracy. The key conclusions of the combined models are summarized as follows:

- **Stacking Ensemble Accuracy:** 82.93% was the accuracy attained by the stacking ensemble, which integrates the predictions from all four models (CatBoost, AdaBoost, XGBoost, and LightGBM). This illustrates how well combining many models may capture intricate patterns in the data.
- **Voting Ensemble Accuracy:** An accuracy of **82.41%** was obtained via a voting ensemble approach, in which the predictions of each model participated equally to the outcome. The performance of this approach is comparable, demonstrating the dependability of the selected models for the categorizing job.

The voting and stacking ensemble approaches both performed well, which qualifies them for use in actual damage from earthquakes assessment applications. But by better using model variety, the stacking ensemble achieved higher accuracy and surpassed the voting method.

VII. CONCLUSION

This study uses CatBoost, AdaBoost, XGBoost, and LightGBM models to forecast earthquake-induced building damage using an ensemble learning technique. When combined with ensemble voting and stacking techniques, these algorithms showed good predictive accuracy. This study will ultimately look into advanced feature engineering approaches, such adding more specific information on age, construction materials, and geographic circumstances, to improve the model's anticipated accuracy. Additionally, adding geographic data such as fault line proximity and soil composition might enhance the model's regional applicability.

Future studies might focus on techniques like SHAP values that improve the understanding of the combined models in order to better understand key damage factors. By expanding its application to real-time catastrophe assessment and looking at transferable lessons for cross-region and cross-disaster scenarios, the model's impact might be further enhanced.

REFERENCES

[1] Yutao Li 1 , Chuanguo Jia 1,2,* , Hong Chen 3,4, Hongchen Su 1 , Jiahao Chen 1 and Duoduo Wang 1 Machine Learning Assessment of Damage Grade for Post-Earthquake Buildings: A Three-Stage Approach Directly Handling Categorical Features

[2] Aishwarya Kumaraswamy, Bhargava N Reddy, Rithvik KollaJ. Richter's Predictor: Modelling Earthquake Damage Using Multi-class Classification Models

[3] Liudmila Prokhorenkova1,2 , Gleb Gusev1,2 , Aleksandr Vorobev1 , Anna Veronika Dorogush1 , Andrey Gulin1 CatBoost: unbiased boosting with categorical feature

[4] * Yoav Freund and Robert E. Schapire A Decision-Theoretic Generalization of On-Line Learning and an Application to Boosting

[5] Guolin Ke1 , Qi Meng2 , Thomas Finley3 , Taifeng Wang1 , Wei Chen1 , Weidong Ma1 , Qiwei Ye1 , Tie-Yan Liu1) LightGBM: A Highly Efficient Gradient Boosting Decision Tree

[6] Mohammad Mahdi Salehi, Seyedali Mirjalili, and Amir Mosavi Building Damage Detection using Ensemble Machine Learning Models with Remote Sensing Data

[7] B. Venkatesh, J. Anuradha A Review of Feature Selection and Its Methods.

[8] Dorogush A, Ershov V, Gulin A CatBoost: Gradient Boosting with Categorical Features Support

[9] Yong Zhang, Hongrui Zhang, Jing Cai, Binbin Yang A Weighted Voting Classifier Based on Differential Evolution

[10] Ke G, Meng Q, Finley T, Wang T, Chen W, Ma W, Ye Q, Liu TY LightGBM: A Highly Efficient Gradient Boosting Decision Tree

[11] Tianqi Chen, Carlos Guestrin XGBoost: A Scalable Tree Boosting System.

[12] Ruihu Wang AdaBoost for Feature Selection, Classification and Its Relation with SVM* , A Review