Applications of and considerations for using machine learning and deep learning tools in earthquake engineering, with focus on soft story building identification

Abhineet Gupta, Todd Macdonald, and Debbie Weiser

Abstract

Machine learning and deep learning technologies are being increasingly used in applied fields including earthquake engineering, with example applications including predicting damage levels from imagery or forecasting earthquakes from acoustic signals. In this study, we present one application of machine learning and deep learning in order to identify soft story buildings that are highly susceptible to collapses during earthquakes. Our two-step model is able to estimate soft story buildings with high recall, from among all buildings in a city. We discuss some of the limitations of our model. We also present that traditional machine learning metrics, like confusion matrices, are not suitable for the problem of damage estimation after an earthquake due to aleatory uncertainties, and present alternative approaches for evaluating these models.

Introduction

Both machine learning (ML) and deep learning (DL) are emerging technologies that are being widely used in many fields including earthquake sciences to get insights from data. We distinguish ML as consisting of algorithms like linear regression and support vector machines involving structured data like numeric or categorical data, while DL as algorithms like neural nets involving non-structured features like images. Development of ML and DL models relies on training data to identify the parameters of the models, which makes these models ideal for modeling complex and sometimes non-intuitive behavior of data. This differs from traditional models that rely on more accurately modeling physics of the process, limited empirical data or expert opinion. However, because of ML and DL models' reliance on data, it is critical that the models are carefully evaluated to assess their performance in modeling the process, especially outside the feature domain of the data.

Deep learning has recently been used in two kaggle.com challenges - 1) to differentiate between damaged and undamaged buildings from images, and 2) to predict upcoming laboratory earthquakes using time series data. Some of the other application of machine learning are rapid prediction and updating of earthquake damage based on real-time ground data, social media data or images; pre-earthquake identification of building features that affect structural vulnerability; and identifying areas with heavy damage from remote sensing.

Case study - soft story identification

A soft story wood-frame building is a structure where the first story is substantially weaker and more flexible than the stories above due to lack of walls or frames at the first floor, leaving the building highly vulnerable to damage in an earthquake. Several cities including San Francisco and Los Angeles have passed mandatory soft story retrofit ordinances to reduce their vulnerability in future earthquakes. One of the first tasks in implementing such a program is to create an inventory of these buildings which can be labor-intensive and expensive. In this project, we use ML tools to rapidly identify soft story buildings for the city of San Jose, CA using building features like construction year and material and Google StreetView imagery.

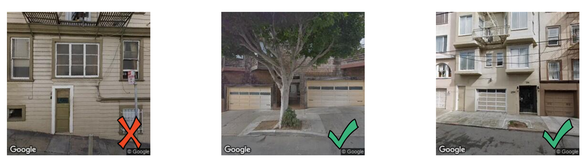
Data

1. Numerical data - based on building attributes of construction type, stories, construction year, foundation type, land use, area, property value, number of bathrooms and total rooms

City	Total buildings	Soft story
	Training/Validation/Testing	
Mountain View	24572/3509/6945	237/45/80
San Francisco	95417/13583/27373	3038/481/854
City in CA	256886/36209/73900	4917/638/1335

2. Image dataset - Google StreetView images from numerical dataset, hand tagged to select images with visible soft stories

Non soft story	1718	Soft story	859



Methodology

1. Stage 1 - Implement Random Forest model optimized to identify soft story buildings with high recall to ensure maximum coverage
   - Reduced buildings of interest from ~200k to ~18k
2. Stage 2 - Implement ResNet-50 architecture Computer Vision model on buildings selected in Stage 1
   - Used image augmentation techniques of horizontal flips, width and height shifts, and zooming
   - Reduced buildings of interest from ~18k to ~3k
3. Stage 3 - Hand label images from Stage 2 and feed back into Computer Vision model for retraining
   - Increased training set from 1718 to 3664 buildings
4. Stage 4 - Hand label additional images identified in Stage 3 as soft story to achieve high precision
   - Identified ~1400 buildings as soft story in the city

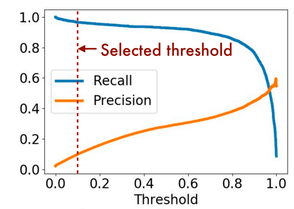
Results and key metrics

Random Forest model

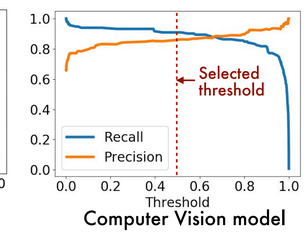
Test set	F1 score	Recall	Precision
SF	0.364	0.950	0.225
Mountain View	0.424	0.962	0.272
City in CA	0.349	0.930	0.216
Overall	0.357	0.938	0.221

Computer Vision model

F1 Score	Precision	Recall	Accuracy
0.884	0.862	0.908	0.880



Random Forest model



Computer Vision model

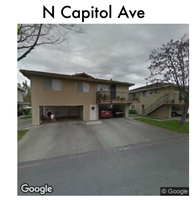
Results provided to the project city

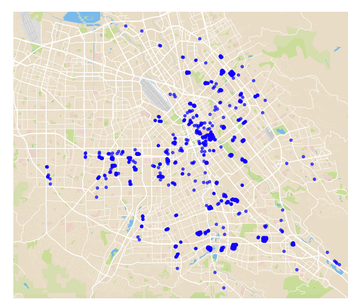
Sample of identified buildings with StreetView images

Saratoga Ave

Stearman Ln

N Capitol Ave





Map of identified soft story buildings

Limitations

- We only identified buildings which visibly appeared to have a soft story via StreetView images and may have missed a subset.
- The results may not generalize well to other cities. Stage 3 of the model with data feedback can improve model performance in a new city. Performing leave-one-city-out cross validation can provide an estimate of this variability.
- Since our models are tuned for high recall, it is necessary to hand label the final results which could be quite costly for larger datasets.

Next Steps

To validate results from the project, we are collecting on-the-ground samples for buildings with and without soft stories. Using this data, we

- can estimate the accuracy, precision and recall for the results of this project on the new city, and
- can improve our ability to identify soft story buildings in the future by creating a repository of new training data and images.

Considerations for creating test sets

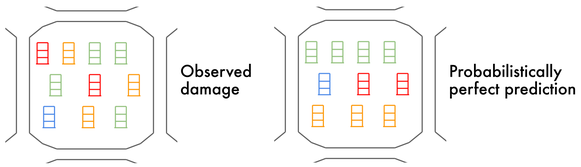
Test sets are used in machine learning to evaluate the generalizability of the models, i.e., their ability to make predictions on unseen data. Generally, test sets are generated by randomly sampling a subset of the available data. However, when data is scarce or does not cover the entire feature domain, for example being from a few cities or events, it is critical to create test sets for rigorous evaluation of model generalizability. Test sets may be improved by -

- holding out spatial regions to evaluate model performance in geographies not available in training,
- holding out temporal regions, to evaluate model performance in future events,
- holding out feature domains to evaluate model's ability to extrapolate, or
- holding out enough data points that metrics can be evaluated to sufficient precision.

Considerations for developing performance metrics

Performance metrics are used to evaluate how well models meet the objectives of the problem.

- Mean square error and accuracy are the most common metrics for regression and classification problems, respectively.
- For classification of extreme events like collapse of a building, optimizing for recall is ideal since it ensures that critical events are not missed.
- For probabilistic forecasts or classification problems with aleatory uncertainty, scoring rules provide a better alternative than accuracy since they compare the probabilistic forecast with the observations.



Conclusions

Machine learning and deep learning are promising techniques that can be used to gain insights from data when it is readily available. Here, we demonstrated their abilities to identify soft story buildings that are especially vulnerable during earthquakes. However, when employing these techniques, it is critical to define the objectives and limitations of the models, for example, street imagery cannot be used to identify those soft story buildings that have parking garages along the back of the building. We can improve the generalizability of these models by carefully developing test sets and identifying metrics that best capture the problem objectives. We described some techniques for test set development and metric identification. By clearly describing data and model limitations, machine learning techniques can be used to solve many labor-intensive and expensive problems while ensuring success for all involved stakeholders.