CP 217: Machine Learning for Cyber-Physical Systems

August-Dec Semester 2024, Project 1 Spec

Competition closes on Sunday 11:59PM IST, October 6th 2024
Report due on Monday 11:59PM IST, October 7th 2024

Competition Link: `https://www.kaggle.com/t/95341718eacc45de92f00e71675e5924`

Weight: 20% of final mark

Introduction

Disaster risk management is a global priority, with earthquakes being a significant cause of casualties and economic losses over the past several decades. The impact of an earthquake on a region depends on various factors, including the structural integrity of buildings, which is critical in reducing the likelihood of collapse. One key aspect of evaluating seismic vulnerability is assessing the ability of buildings to sustain earthquake loads. Multi-story structures with abrupt changes in story stiffness are particularly prone to collapse, making it essential to rapidly and accurately identify potential structural deficiencies across large building inventories.

Recent advances in data availability have opened new opportunities for improving seismic risk assessment. Street-view images, such as those from Google Street View (GSV), provide valuable visual information that can be leveraged to identify building types and materials. Such information is essential for developing accurate earthquake exposure models, which help estimate the vulnerability of buildings in a given area. By automating the analysis of these images, machine learning (ML) models can provide a faster and more scalable solution for identifying structural characteristics relevant to earthquake preparedness.

Your Task

Your objective in this project is to develop an automated ML model to identify building types/materials from street-view images. You may either develop a novel ML approach or utilize existing models that have been publicly released for similar datasets, as long as their use is properly justified and acknowledged. Additionally, you are encouraged to introduce innovative techniques or methods to improve existing models. An extra mark will be awarded for successfully incorporating such an approach, provided it was not covered in class or workshops and is properly justified in your report.

To make the project fun, we will run it as a Kaggle in-class competition. Kaggle is one of the most popular online platforms for predictive modelling and analytics tasks. You will be competing with other students in the class. The following sections give more details on data format, the use of Kaggle, and marking scheme. Your assessment will be based on your final ranking in the competition, the absolute score that you achieve, and your report. The marking scheme is designed so that you will pass if you put in effort. So fear not and embrace the power of machine learning!

Data Format

You will have access to three types of folders/files, primarily train_data folder, test_data folder, and sample_submission.csv file. **These folders/files are available in a Zip file in Class Team under "Project 1 Data and Specification" Channel in "File" section.**

The train_data folder consists of five subfolders, each representing a different building class: A, B, C, D, and S. Detailed descriptions of these classes can be found in the Table 2 at the bottom of this PDF. Each subfolder contains images of buildings that correspond to a specific type or material. Each image within a class folder is named using the

format: <image_number>_<building_class>.jpg. The building classes, represented by letters, should be converted into numerical values as follows: A $\rightarrow$ 1, B $\rightarrow$ 2, C $\rightarrow$ 3, D $\rightarrow$ 4, S $\rightarrow$ 5 . The entire train_data set consists of 2,516 images distributed across these five classes.

The test_data folder contains 478 images spanning the same five building classes. In this folder, the images are named sequentially using Image IDs, but the class information is not provided. Your task in this project is to predict the correct class for each of these images.

A sample_submission.csv file shared with you shows an example submission file. The sample_submission.csv file contains the same IDs in its first column as those found in the test_data folder, listed in the same order. Once you have done predictions for each ID in the test.txt file using your own ML model, you should create a submission file in CSV format, similar to the sample_submission.csv, with the following structure.

```
ID,Predictions
1, 3
2, 4
3, 3
...
```

The first line should be a header, exactly as shown in sample_submission.csv. There should be 478 rows (excluding the header row) in total, each with a unique ID. The seconds column in the sample_submission.csv corresponds to the predictions (which are given randomly). The IDs of predictions should match the IDs of images in the test_data folder, in the same order.

Note that you **should not** inspect the test data closely or hand label the data in the submission file by just inspecting the test data, as this is cheating and compromises the point of the project (though please inspect your submissions to ensure your files are in the right format, of the right size, etc.). **Note that you will have to provide your code in your submission (discussed later) and we will run it at our end to ensure you have not done any such cheating.**

The test set will be used to generate the classification accuracy for your performance. During the competition, the classification accuracy on a subset of the test set will be used to rank you in the leaderboard. As we provide no explicit validation set, you may want to reserve part of the training partition for this purpose during model development. Your job is to develop an algorithm that can automatically capture the nuances of the problem, in order to generalise well to unseen data (estimated here over the test set.)

Kaggle In-class Competition

You can join this competition only using the following link.
Link: `https://www.kaggle.com/t/95341718eacc45de92f00e71675e5924`
Please do the following by the end of the first week after receiving this assignment:

- Setup an account on Kaggle with username and email being your IISc student email.
- Form your team of student peers (Note that some or all teams may be formed by the Course Instructor to make sure each team has a student with some prior experience with programming)
- Connect with your team mates on Kaggle as a Kaggle team, using a team name. You can choose any team name e.g., Shaktimaan, Spyderman etc. Only submit your entries via the team; and
- Register your final team using the Google Form. Link: `https://forms.gle/Q7UKBdJTfzJS4LiJ8`). There are three sections in this form. First section asks your basic details. The second section is for those who have already decided/ registered a team on Kaggle competition. The last section is for those who are looking for a team member(s) to form a team.

Teams should consist of three individuals. If you cannot find a team, please introduce yourself to fellow students in workshop or the lecture, or post to *Project 1 Discussion Channel* in Class Team that you are in search of a team. Feel free to mention your skills/strength and what you are looking for in other members. In only very rare occassions will we permit teams of less than three (and we will mark all teams based on our expectations of what a team of three could achieve). The motivation for working in teams is that in industry, practising machine learning experts work effectively in teams. You should only make submissions using the team name, individual submissions are not allowed and may attract penalties. Note that teams will be limited to 3 submissions per day

The real labels for the test data are hidden from you, but were made available to Kaggle. Each time a submission is made, half of the predictions (50% of the test data) will be used to compute your public score and determine your rank in the public leaderboard. This information will become available from the competition page almost immediately. At the same time, the other half of predictions is used to compute a private accuracy and rank in private leaderboard, and this information will be hidden from you. At the end of the competition, only private scores and private ranks will be used for assessment and will be revealed publicly. This type of scoring is a common practice and was introduced to discourage overfitting to public leaderboard. A good model should generalize and work well on new data, which in this case is represented by the portion of data with the hidden accuracy.

The evaluation score used in this competition is the classification accuracy over all classes, defined as the number of instances labelled correctly as a fraction of the total number of instances. Before the end of the competition, each team will need to choose 3 best submissions for scoring. These do not have to be the latest submissions. Kaggle will compute a private accuracy for the chosen submissions only. The best out of the 3 will then be automatically selected and this private score and the corresponding private leaderboard ranking will be used for marking.

Each participant can do a maximum of 3 submissions every day. Before the end of the competition, each of you will need to choose your 3 best submissions for final scoring. These do not have to be the latest submissions. Kaggle will compute a private accuracy for the chosen submissions only. The best out of the 3 will then be automatically selected and this private score and the corresponding private leaderboard ranking will be used for marking. If you don't choose any submission, Kaggle will, by default, consider your best submission performance on the public leaderboard for computing the private accuracy.

Report

Each team (only one from the team) will submit a report with the description, analysis, and comparative assessment (where applicable) of the method or methods used. There is no fixed template for the report, but it should provide the following sections:

1. A Kaggle team name with all the team members' name should be mentioned at the beginning of the project. You can also mention your final ranking on the private leaderboard.

2. A very brief description of the problem and introduction of any notation that you adopt in the report.

3. A brief description on any data pre-processing, validation, sampling, class distributions, etc., you tried. Any interesting details about software setup or your experimental pipeline, and any problems you encountered and what you learned. In many cases these issues are at least as important as the learning algorithm, if not more important.

4. Description of your final approach(s), the motivation and reasoning behind it, and why you think it performed well/not well in the competition.

5. Any other alternatives you considered and why you chose your final approach over these (this may be in the form of empirical evaluation). Reflect on why the method(s) performed or didn't perform well. If you tried

different models or different hyperparameters, compare the methods to each other in the context of this competition. Your reasoning can be in the form of empirical evaluation, but it must be to support your reasoning (examples like "method A with X features and Y value of parameter for accuracy 0.60 and method B, got accuracy 0.7, hence we use method B", with no further explanation, will be marked down).

6. If you used any feature transformations, selected only some useful features, or generated new features, you should also describe them in the report along with the expected effect from using such features and effect observed after implementation and evaluation. In comparing methods, you may want to use an evaluation besides measuring accuracy, in order to better understand the kinds of mistakes being made (e.g., with rare (minor) classes.)

Your description of the algorithms should be clear and concise. You should write it at a level that a postgraduate student can read and understand without difficulty. If you use any existing algorithms, you do not have to rewrite the complete description but must provide a summary that shows your understanding and references to the relevant literature. In the report, we will be very interested in seeing evidence of your thought processes and reasoning for choosing one algorithm over another.

The report should be submitted as a PDF, and be no more than three *A*4 pages of content, including all plots, tables and references[1] (single column, font size of 11 or more and margins at least 1 cm, much like this document). You do not need to include a cover page. **If a report is longer than three pages in length, we will only read and assess the report up to page three and ignore further pages.**. Marks may be deducted if these instructions are not followed.

Submission and Assessment

In summary, each student is required to make the following submissions for this project:

- One or more submission files with predictions for test data (at Kaggle). This submission must be of the expected format as described above, and produce a place somewhere on the leaderboard. Invalid submissions do not attract marks for the competition portion of grading;
- Report in PDF format (via "Assignment" Section for this project in Class Teams);
- Source code used in this project as a single ZIP archive (via "Assignment" Section for this project in Class Teams). Your code can be in any of the following languages C, C++, Python, Jupyter Notebook, R or MATLAB. If there is another language you like to use, please contact us first. If the language requires compiling, a makefile or script must be provide to build the executables. We may or may not run your code, but we will definitely read. You should not include the training or test data file in the ZIP file.

The project will be marked out of 30. No late submission of Kaggle portion will be accepted; late submissions of reports will incur a deduction of 3 marks per day, or part thereof. Based on our experimentation with the project task and the design of the marking scheme below, we expect that all reasonable efforts at the project will achieve a passing grade or higher. So relax and have fun!

Marking Scheme

Kaggle competition (15 marks) This mark takes into account both achieved accuracy, as well as your standing in the class. Assuming N is the number of students, and R is your rank in the class, the mark you get for the competition

[1] Plots can be useful for model selection, assessing convergence, features importance, displaying results and model interpretation, among other things. For instance, plotting the parameters of your model with respect to the objective function can often give insights into what the model has learned.

part is

$$12 \times \frac{\max\{\min(acc, 0.90) - 0.20, 0\}}{0.70} + 3 \times \frac{N - R}{N - 1}$$

The first term constitutes up to 12 marks, and rewards high accuracy systems with a maximum score for excellent systems with $\geq$ 90% accuracy, and zero score to those with scores $\leq$ 20% which are just little better than random guessing. The second term, worth 3 marks, is based on your rank and is designed to encourage competition and innovation. Ties are handled so that you are not penalised by the tie. All who are tied will get the same marks for score, but ranking will be decided based on total number of submission entries. The score with fewer entries will be ranked higher among tied ones.

External teams of unenrolled students (auditing the subject) may participate, but their entries will be removed before computing the final rankings and the above expression, and will not affect registered students' grades. Note that invalid submissions will come last and will attract a mark of 0 for the score, so please ensure your output conforms to the specified requirements, and have at least some kind of valid submission early on!

Report (15 marks) The report will be marked using the rubric in Table 1.

Bonus Mark (1 mark) you will get 1 bonus mark if you have used any ML model which was not taught in the classes/workshops before the submission deadline, or if you have used any innovative techniques in that improves your model performance on test data or you achieved the best performance using any non-neural network model. You need to provide this information in your report with proper justification, to get this 1 bonus mark.

Plagiarism policy

You are reminded that all submitted project work in this subject is to be your own individual work. Automated similarity checking software will be used to compare submissions. It is University policy that cheating by students in any form is not permitted, and that work submitted for assessment purposes must be the independent work of the student(s) concerned. For more details, please see the policy at `https://iisc.ac.in/about/student-corner/academic-integrity/`.

Critical Analysis (8 marks)	Report Clarity and Structure (7 marks)
7–8 marks Final approach is well motivated and its advantages/disadvantages clearly discussed; thorough and insightful analysis of why the final approach works/not work for provided training data; insightful discussion and analysis of other approaches and why they were not used	*6–7 marks* Very clear and accessible description of all that has been done, a postgraduate student can pick up the report and read with no difficulty
5–6 marks Final approach is reasonably motivated and its advantages/disadvantages somewhat discussed; good analysis of why the final approach works/not work for provided training data; some discussion and analysis of other approaches and why they were not used	*4–5 marks* Clear description for the most part, with some minor deficiencies/loose ends (e.g., there are no- table gaps and/or unclear sections)
3–4 marks Advantages/disadvantages discussed; limited analysis of why the final approach works/not work for provided training data; limited discussion and analysis of other approaches and why they were not used	*2–3 marks* Generally clear description, but there are notable gaps and/or unclear sections.
1–2 marks Final approach is barely or not motivated and its advantages/disadvantages are not discussed; no analysis of why the final approach works/not work for provided training data; little or no discussion and analysis of other approaches and why they were not used	*1 mark* The report is unclear on the whole, omits all key reference,and the reader can barely discern what has been done

Table 1: Report marking rubric.

Class type	Building Material	Other Properties
A (1)	Steel Buildings	Extensive presence of glass or reflective cover, rounded surfaces, exposed steel skeleton, complex frame geometry, presence of slabs, significant slenderness
B(2)	Concrete	Exposed concrete, thick and regular structure skeleton, boxy aspect, smooth texture, persistent opening arrangement
C (3)	Masonry	Reinforced or Unreinforced, bricks exposed, low opening/wall ratio, no soft story or large openings in bottom part, present of decorative patterns
D (4)	Wooden framed	Wooden frame exposed, wooden bow windows, irregular opening structure, slanted roof, vivid colouring and painting outlines
S (5)	Steel with panel buildings	typical for industrial or retail, max 2 floors (rows of openings), big and sparse lower level openings, non slanted roof, no elaborate texture on the outside, panel by panel cover

Table 2: Building class: Material and Properties.