

## CSC 180-01 Intelligent Systems (Fall 2024)

### Project 3: House Price Prediction using TensorFlow Functional API

**Due at 10:30 am, Wednesday, October 23, 2024**

**Demo: Wednesday, October 23, 2024**

Note that you must **print and fill in** your names on the Evaluation Form and bring the copy to your demo session to receive credit.

#### 1. Problem Formulation

Most of existing house price estimation systems rely only on textual data like its neighborhood area and the number of rooms. In this project, we practice with TensorFlow Functional API to extract visual features from house photos and combining them with the house's textual information.

The goal of this project is to build **a novel house price estimation system by using both textual and visual inputs**, other than only using textual information such as area, neighborhood, and number of bedrooms.

Build your best deep learning model and then show its RMSE and lift chart on test data.

#### 2. Dataset

<https://github.com/emanhamed/Houses-dataset>

This dataset contains both visual and textual information.

- Each house is represented by four images for bedroom, bathroom, kitchen and a frontal image of the house.
- The dataset folder contains 2140 images, 4 images for each house.

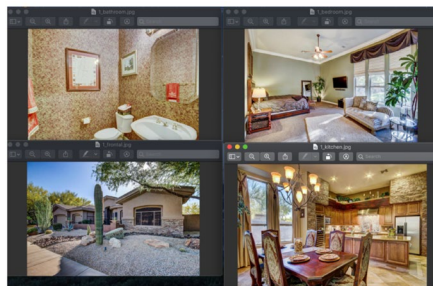
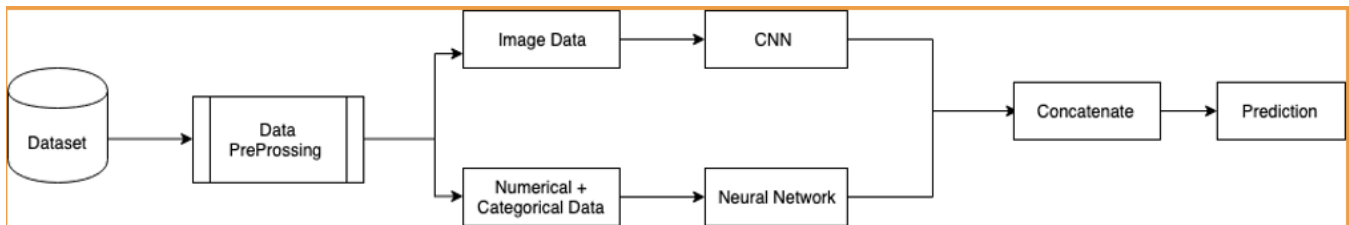


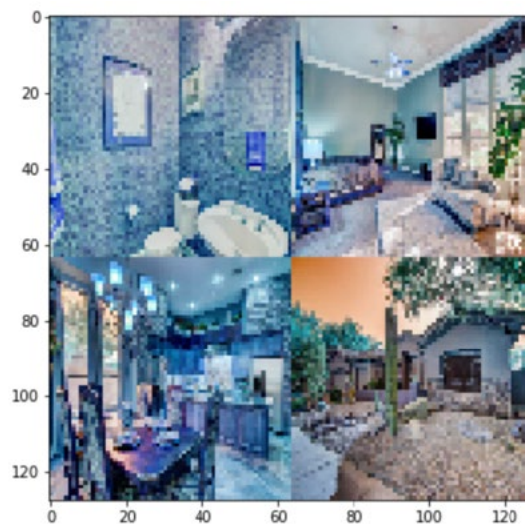
Fig. 1 Example of Images of house

- The dataset has a text file that contains the textual metadata of the dataset. Each row in the file represents the number of house in order. The numbers represent number of bedrooms, number of bathrooms, area of the house, zipcode and the price.

### 3. Model Design



We will use TensorFlow Functional API. For each house, for simplicity, let's create a combined image using the 4 house images (bedroom, bathroom, frontal and kitchen, as shown below). You can find the code to create such images in the cheat sheet on the same Canvas page. To speed up training, you may use Google Colab GPU.



### 4. Requirements

- To remove the outliers first. For example, you only keep the houses with a price between 100K and 900K.
- Split data for training and testing. Use training data to train your models and evaluate the model quality using test data

- Encode categorical features and normalize numeric features.
- You must use EarlyStopping when training neural networks using Tensorflow.

## 5. Grading Breakdown

You may feel this project is described with some certain degree of vagueness, which is left on purpose. In other words, **creativity is strongly encouraged**. Your grade for this project will be based on the soundness of your design, the novelty of your work, and the effort you put into the project.

Use [the evaluation form on Canvas](#) as a checklist to make sure your work meets all the requirements.

## 6. Teaming

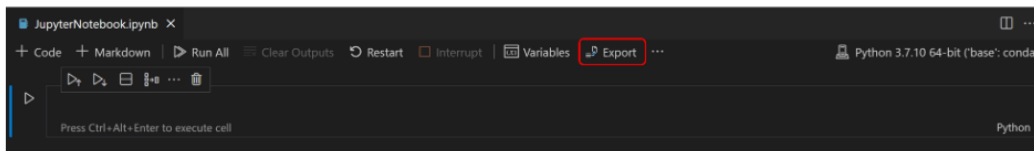
Students must work in teams of at most 4 people. Think clearly about who will do what on the project. Normally people in the same group will receive the same grade. However, the instructor reserves the right to assign different grades to team members depending on their contributions. So you should choose partner carefully!

## 7. Deliverables

- (1) The **HTML version of your notebook** that includes all your source code.

### Export your Jupyter Notebook

You can export a Jupyter Notebook as a Python file ( `.py` ), a PDF, or an HTML file. To export, select the Export action on the main toolbar. You'll then be presented with a dropdown of file format options.



5 pts will be deducted for the incorrect file format.

**NOTE: Comment each notebook cell with author name and date.**

- (2) **Your report in PDF format**, with your name, your id, course title, assignment id, and due date on the first page. As for length, I would expect a report with more than one page. Your report should include the following sections (but not limited to):

- Problem Statement

- Methodology
- Experimental Results and Analysis
- Task Division and Project Reflection

In the section “Task Division and Project Reflection”, describe the following:

- who is responsible for which part,
- challenges your group encountered and how you solved them
- and what you have learned from the project as a team.

(3) A **separate text file** named “additional.txt”, which describes the additional features you implemented.

All the deliverables must be submitted **by team leader** on Canvas before

**10:30 am, Wednesday, October 23, 2024**

NO late submissions will be accepted.

## 8. Hint:

Cheatsheet is provided for data/image preprocessing.

## 9. Additional Features

- Can you try transfer learning? <https://keras.io/api/applications/>  
There are some models you may want to consider
  - VGG16
  - ResNet50
  - MobileNetV2
- Does it help if we treat four images separately without combining them together as input?
- Is there a better way to take advantage of zipcode? Treat zipcode as a separate input channel? Remove zipcode without enough houses?
- A research paper on the same dataset: *Eman H. Ahmed, Mohamed N. Moustafa, House price estimation from visual and textual features.* <https://arxiv.org/pdf/1609.08399.pdf> Can you learn anything from the paper to improve your model?