[redacted]

Project 1: Yelp Business Rating Prediction using Tensorflow

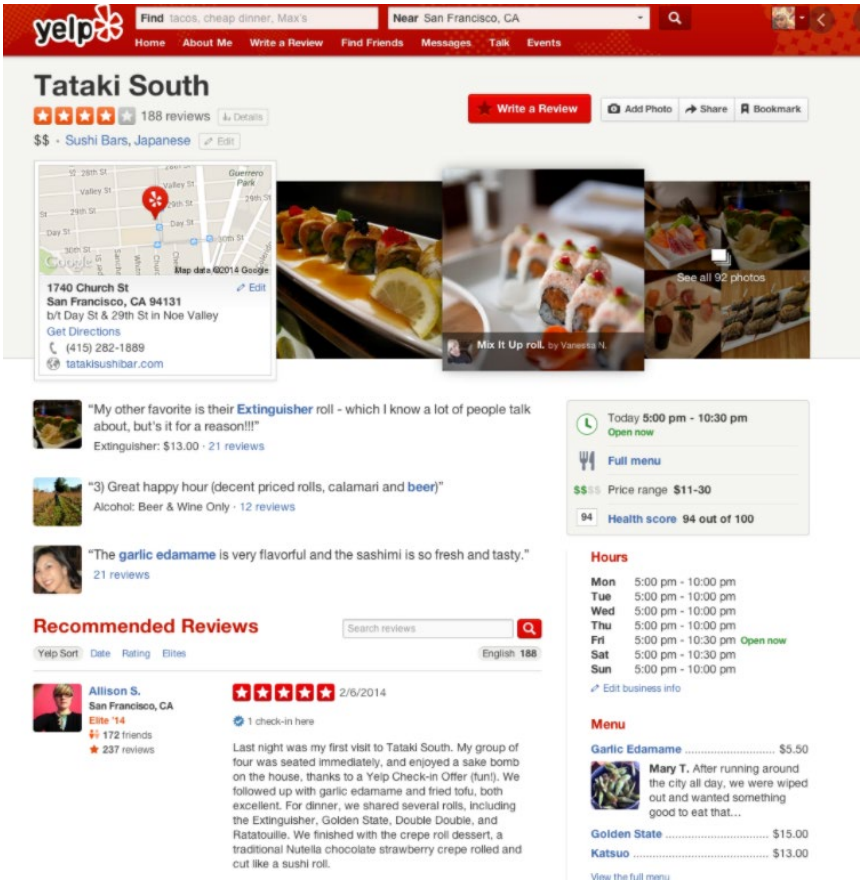
[redacted]

1. Problem Formulation

In this project, we aim to predict a business's stars rating based on all the review text for that business using neural network implementations in TensorFlow. Consider this problem as a regression problem.

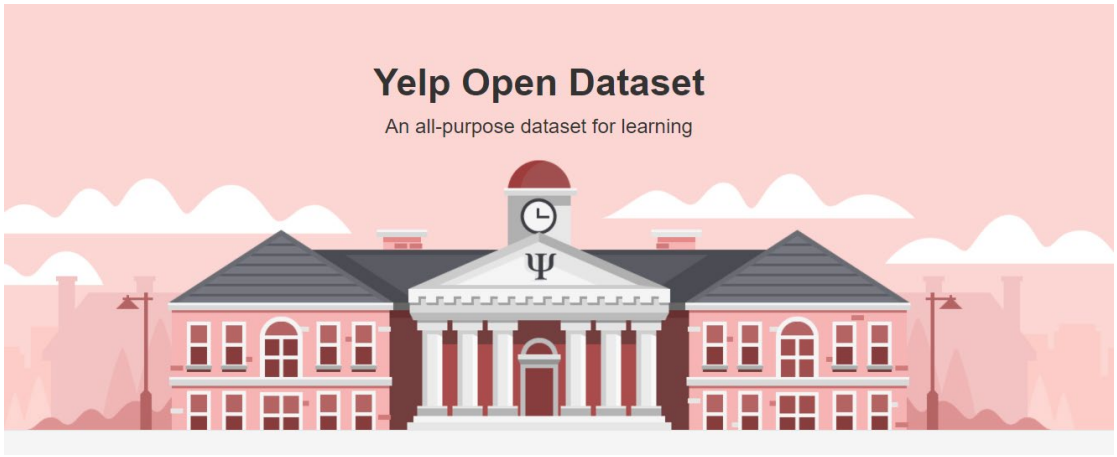
(1) Report the RMSE and plot the lift chart of the BEST neural network model you have obtained.

(2) Choose 3-5 arbitrary businesses from your test data (preferably from different categories). Show the names, the true star ratings, and the predicted ratings (from your best model) of those businesses.

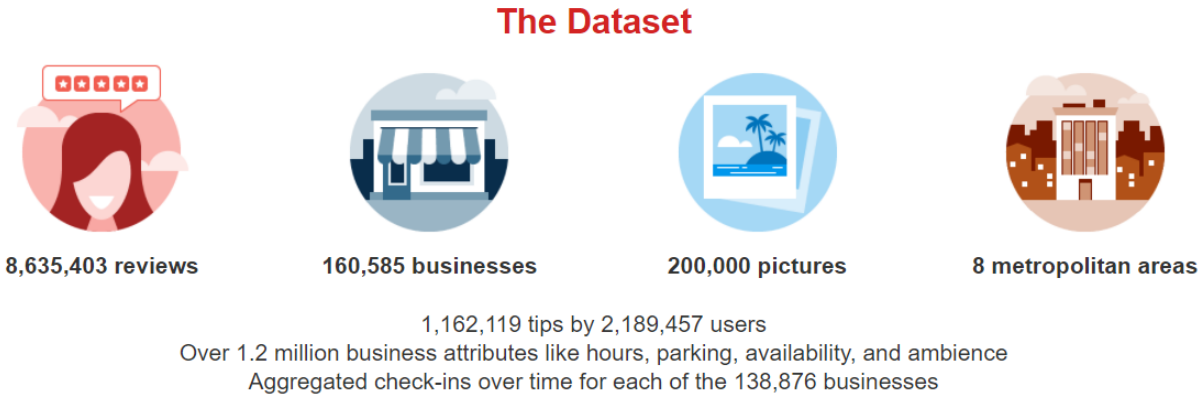


2. Dataset

https://www.yelp.com/dataset



The dataset contains several JSON files. You can find the format of the data here: https://www.yelp.com/dataset/documentation/main



Example file formats are as follows.

business

```
{
    'type': 'business',
    'business_id': (encrypted business id),
    'name': (business name),
    'neighborhoods': [(hood names)],
    'full_address': (localized address),
    'city': (city),
    'state': (state),
    'latitude': latitude,
    'longitude': longitude,
    'stars': (star rating, rounded to half-stars),
    'review_count': review count,
    'categories': [(localized category names)]
    'open': True / False (corresponds to closed, not business hours),
    'hours': {
        (day_of_week): {
            'open': (HH:MM),
            'close': (HH:MM)
        },
        ...
    },
    'attributes': {
        (attribute_name): (attribute_value),
        ...
    },
}
```

review

```
{
    'type': 'review',
    'business_id': (encrypted business id),
    'user_id': (encrypted user id),
    'stars': (star rating, rounded to half-stars),
    'text': (review text),
    'date': (date, formatted like '2012-03-14'),
    'votes': {(vote type): (count)},
}
```

3. Data Cleaning

In this project, we will only consider the businesses with at least 20 reviews. So remove all the businesses with less than 20 reviews.

4. Requirements

- You are required to split data to training and test. Use training data to train your models and evaluate the model quality using test data.
- Use TF-IDF to extract features from reviews. If you experience low memory issue when using ***tfidfVectorzier***, set parameters ***max_df, min_df,*** and ***max_features*** appropriately.

- You must use EarlyStopping when training neural networks using Tensorflow.
- Tuning the following hyperparameters when training neural networks using Tensorflow and record how they affect performance in your report. Tabulate your findings.
  - **Activation:** relu, sigmoid, tanh
  - **Number of layers and neuron count for each layer**
  - **Optimizer:** adam and sgd.

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.

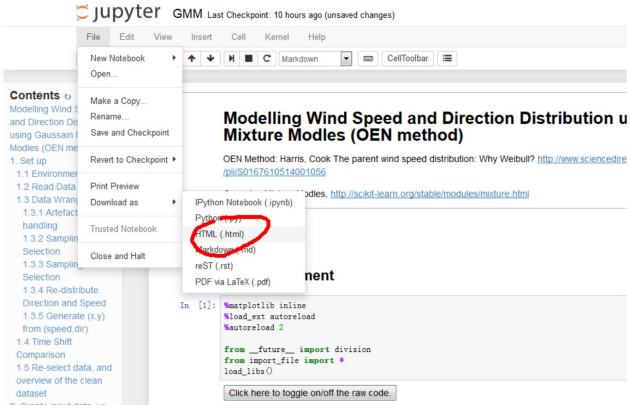
Use the evaluation form on Canvas as a checklist to make sure your work meet all the requirements.

6. Teaming:

Students must work in teams with no more than 3 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 reserve 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. Go to "File" and then "Download as". Click "HTML" to convert the notebook to HTML.



5 pts will be deducted for the incorrect file format.

(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):

(1) Problem Statement
(2) Methodology
(3) Experimental Results and Analysis
(4) Task Division and Project Reflection

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

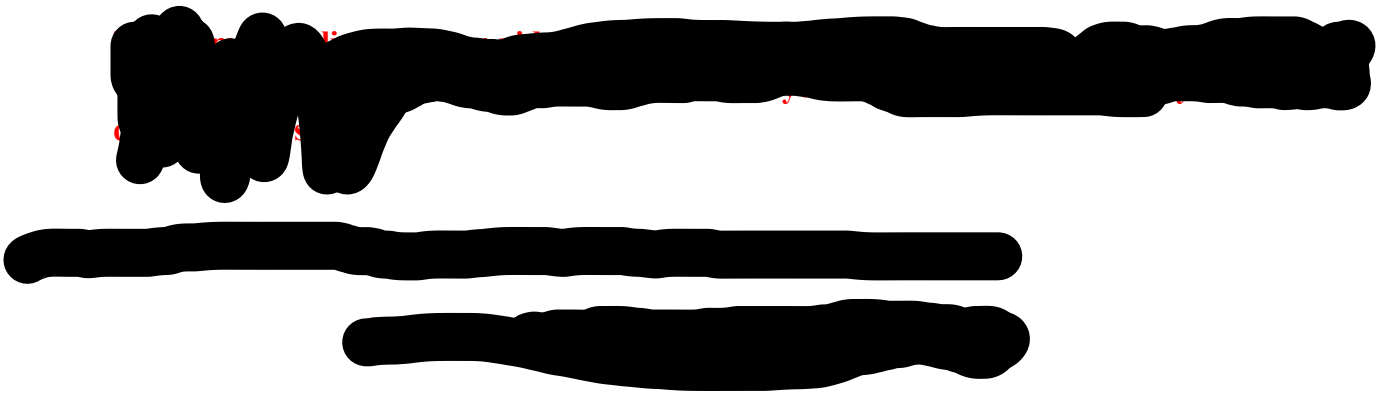
- who is responsible for which part,
- challenges your group encountered and how you solved them
- and what you have learned from the project as a team.

10 pts will be deducted for missing the section of task division and project reflection.

To submit your notebook and the report, go to Canvas "Assignments" and use "Project X (submit your code and report here)". Use the evaluation form on Canvas as a checklist to make sure your work meet all the requirements.

(3) Link to your video presentation shared to the discussion board. Each team have **three minutes** to demo your work. Failure to submit the video presentation will result in **zero** point for the project. The following is how you should allocate your time:

- Model/code design (1 minute)
- Findings/results (1 minutes)
- Task division, challenges encountered, and what you learned from the project (1 minutes)



NO late submissions will be accepted.

8. Peer Review:

During the class after the deadline, please review and comment on the presentations from other teams by replying to their posts. It is a great chance for you to learn from other people's work. Please be nice, and provide constructive, specific feedbacks. You will become a better, more effective learner when you found yourself in a community of active learners!

9. Coding Hints

- You may use the following code to convert JSON data into a tabular format Pandas can read.

```
import json
import csv
import pandas as pd

outfile = open("review_stars.tsv", 'w')
sfile = csv.writer(outfile, delimiter ="\t", quoting=csv.QUOTE_MINIMAL)
sfile.writerow(['business_id','stars', 'text'])

with open('yelp_academic_dataset_review.json', encoding="utf-8") as f:
   for line in f:
      row = json.loads(line)
      # some special char must be encoded in 'utf-8'
      sfile.writerow([row['business_id'], row['stars'], (row['text']).encode('utf-8')])

outfile.close()

df= pd.read_csv('review_stars.tsv', delimiter ="\t", encoding="utf-8")
```

- You may use the following sample code to group ALL the reviews by each business and create a new dataframe, where each line is a business with all its reviews aggregated together. From there, you then use ***tfidfVectorzier*** to obtain TFIDF representation for each business.

```
df_review_agg = df.groupby('business_id')['text'].sum()
```

df_ready_for_sklearn = pd.DataFrame({'business_id': df_review_agg.index, 'all_reviews': df_review_agg.values})

- To align all the reviews of a business with its business star rating, you may want to join the review table with the business table on the business_id column. Pandas supports high performance SQL join operations. Use Pandas function ***pd.merge()*** to **merge (or to say, join) two dataframes** based on values in one particular column. See examples here:

  https://chrisalbon.com/code/python/data_wrangling/pandas_join_merge_dataframe/

- If you want to **merge two numpy arrays**, use Numpy function ***np.concatenate()***

- Convert a Pandas Dataframe to its corresponding Numpy array representation, use **to_numpy()**

- For one-hot coding, you may use Pandas ***pd.get_dummies().***

10. Think beyond the Project

- Can you build a more accurate model by taking the number of reviews (review count) into account?
- What other information can be used to train a more accurate model? Business categories? Check-in count?

- Can you build a more accurate model by focusing only on a particular business category?