CSC 180-01 Intelligent Systems (Fall 2024)

Project 1: Yelp Business Rating Prediction using Tensorflow

Due at 10:30 am, Wednesday, September 25, 2024

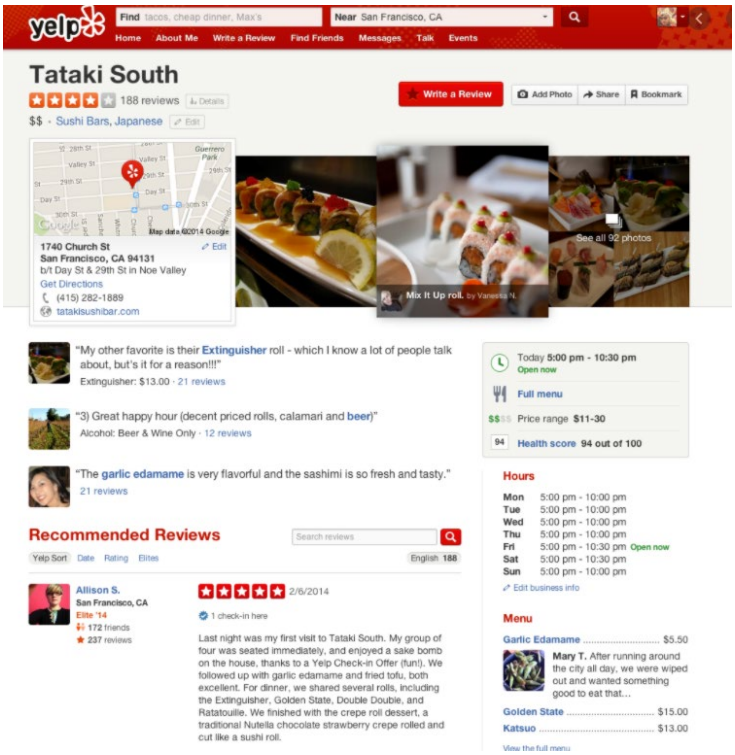
Demo: Wednesday, September 25, 2024

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

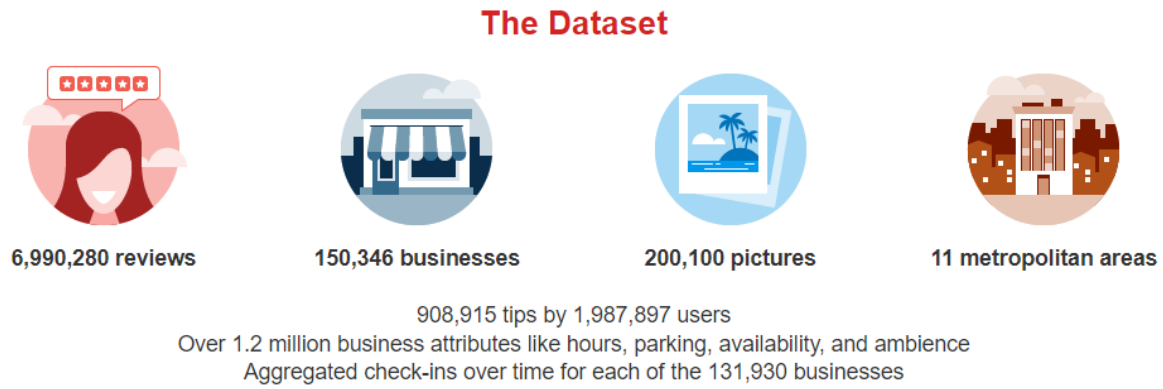
In this project, we aim to predict a business's stars rating based on all the reviews for that business using fully-connected neural network implementations in TensorFlow. Consider this problem as a regression problem.

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

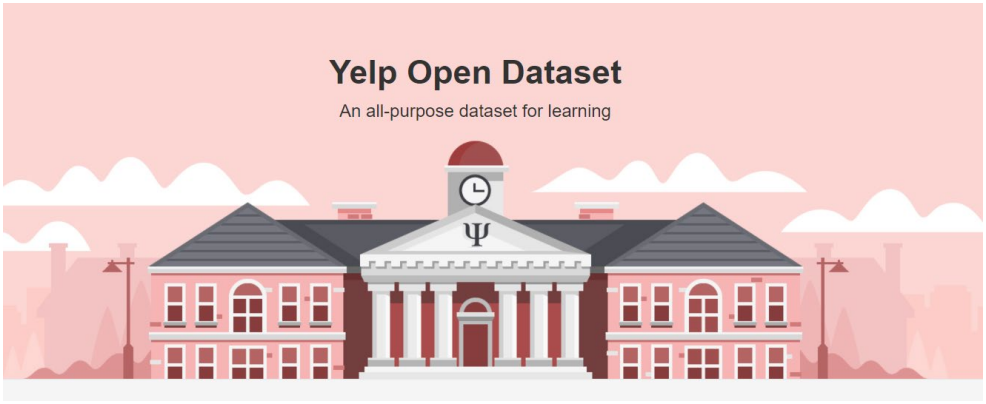


2. Dataset

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. Remove all the businesses with less than 20 reviews. Feel free to use only a subset of business and review data (at least 10K businesses).

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 **tabulate** all the results of each model on how they affect performance in your report. Also, save all the models you have tried as a proof in your notebook.
    - **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.

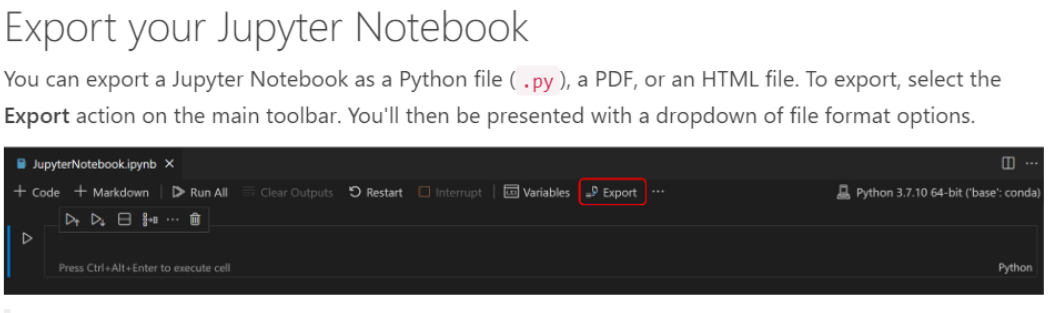
Use the evaluation form on Canvas as a checklist to make sure your work meets all the requirements.

6. Teaming:

Students must work in teams with no more than 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! You are also welcome to work on your own.

7. Deliverables:

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



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.

(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, September 25, 2024

NO late submissions will be accepted.

8. Coding Hints

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

  review = pd.read_json('yelp_academic_dataset_review.json', lines=True, nrows = 1000000)

  business = pd.read_json('yelp_academic_dataset_business.json', lines=True, nrows = 1000000)

- You may use the following 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_to_be_sent_to_sklearn = pd.DataFrame({'business_id': df_review_agg.index, 'all_reviews': df_review_agg.values})

For how to use *tfidfVectorzier*, check here:

https://scikit-learn.org/stable/modules/generated/sklearn.feature_extraction.text.TfidfVectorizer.html

https://medium.com/@cmukesh8688/tf-idf-vectorizer-scikit-learn-dbc0244a911a

- 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.

- 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().***

9. Additional Features

- Can you build a more accurate model by taking other features, e.g., 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? Address?

- Should we create a per-category model for more accurate prediction? Would that model perform better than a generic prediction model?