**Problem Statement**

This project aims to predict Yelp and Google Maps restaurant ratings using photos from customer reviews. We want to determine whether it is possible to predict the quality of a restaurant using only image data as opposed to explicit ratings. This is interesting for multiple reasons. Firstly, many review sites allow users to post images without an associated rating. Generating a rating makes an image-only review considerably more useful for the purposes of determining the relative quality of different businesses. Secondly, the project allows social media platforms with image data to create travel guide or business recommendation features for users.

**Data Gathering Plan**

Data will be gathered using the Yelp and Google Maps APIs as well as the Scrapy web-scraping framework. Both the Yelp and Google Maps APIs function similarly, allowing users to retrieve a list of relevant businesses based on a search query and a location. Repeated calls to the two APIs will be made using combinations of a search query and a location. The list of possible search queries will look like [‘French restaurants’, ‘Chinese restaurants’, ‘Indian restaurants’, etc.] and the list of possible locations will look like [‘Toronto’, ‘Montreal’, ‘Vancouver’, etc.]. The Google Maps API provides a business ID, rating, and up to 10 photographs for each business. The Yelp API provides a place ID, rating and three photographs per business, but an additional 9 photographs can be acquired through web scraping. The business ID and place ID will ensure data on a business is not collected twice on the Yelp and Google Maps APIs respectively. We aim to collect approximately 10,000 photographs, each with an associated restaurant rating. Data augmentation techniques such as cropping, flipping, noise injection may be considered to expand the dataset. All images will be resized to 256x256 before being fed into our CNN.

Yelp API details

1. GET https://api.yelp.com/v3/businesses/search to search for business IDs
2. GET [https://api.yelp.com/v3/businesses/{id](https://api.yelp.com/v3/businesses/%7Bid)} to get restaurant name and rating
3. Scrape <https://www.yelp.ca/biz_photos/scrollable_photos/>{id} for “top” 12 photos

Google Maps API details

* GET [https://maps.googleapis.com/maps/api/place/textsearch/*output*?*parameters*](https://maps.googleapis.com/maps/api/place/textsearch/output?parameters)to search for place IDs, names and ratings.
* GET [https://maps.googleapis.com/maps/api/place/details/*output*?*parameters*](https://maps.googleapis.com/maps/api/place/details/output?parameters)to get 10 photographs of the business.

As a backup plan or to further expand the dataset, Yelp provides a Photos dataset [<https://www.yelp.com/dataset/documentation/photos>] which includes 200,000 photos with associated business IDs. The Yelp API can be used to get the rating for each of the businesses.

**Prior (and More Prior) Work**

[**https://docs.google.com/document/d/1pjkeaFcKeaCODnnppRKzoOzNZm2PMImuWkYla2ttCTI/edit**](https://docs.google.com/document/d/1pjkeaFcKeaCODnnppRKzoOzNZm2PMImuWkYla2ttCTI/edit)

The paper is a good pick for the topic. The summary is clear and evinces understanding of the paper and its significance. The important and relevant points from the paper are pointed out in the summary.

**Technical Methods (hshmat + vince)**

From <https://www.ijcai.org/Proceedings/15/Papers/193.pdf>

**Their methodology:**

* **USWCVM** (user-word composition vector model) and **DCVM** (document composition vector model) used here
  + Processes document (the review) to find representation in semantic space
  + First starts low-level with word vectors and fed into holistic DCVM architecture
  + This representation is then fed into a metric label predictor (still not sure what it does, honestly)
* Metric labeling of USWCVM + DCVM output used to then produce a rating based off of the text review

**Can we leverage this architecture but using images instead of text?**

* What if we used a CNN that translates images -> semantic representation?
  + We would be able to replace the USWCVM and DCVM with this architecture instead
  + <https://arxiv.org/pdf/2106.11596.pdf>
    - May be overkill: multilayer semantic representation with multi-label images
    - Performs supposedly better (apparently most CNNs leave translation to semantic space to the last layer)
  + <https://static.googleusercontent.com/media/research.google.com/en//pubs/archive/41473.pdf>
    - Translates semantic knowledge in the text domain for application to images
    - Requires pre-training with a language model (we can make use of text reviews we know are available)
    - Post-training, NN only needs image inputs for deployment
* Architecture: image as input -> CNN gets semantic representation -> semantic representation fed to metric label predictor -> rating as output

**Do we need to bother with semantic representation at all?**

Initial thoughts…

* Assumption: we *will* be using a CNN since it’s the most conventional/effective way to process images
* Seems as though deep CNNs have a lot of overlap with text/audio processing -> semantic space to begin with
  + <https://towardsdatascience.com/an-introduction-to-convolutional-neural-networks-eb0b60b58fd7#:~:text=The%20most%20common%20use%20for,which%20CNNs%20perform%20well%20at>. (probably not the most reputable source, but a good summary of CNN uses + fields of efficacy)
* So if we were to use a generic deep CNN with the goal of going from image -> representation…
  + would some hidden layers of the CNN extract/process/output semantic features anyways?
  + And if that’s the case, would we see better performance having part of our architecture be specialized for that?

**Anticipated Results**

* How good is our NN gonna be???
* **Determine after reading papers**

**Some random notes:**

* **Might be good to have a NN architecture diagram to indicate a baseline model**
* **Might be good to actually do some data gathering on Yelp for proof of concept**
* **Some research that this is a good problem- not sure how to get this info tho…**
* **Discuss how our problem is similar enuff to previous to borrow some ideas but different enuff to allow for innovation and a problem of actual interest**

**Questions (+ answers)**

* How do we consolidate GMaps vs Yelp reviews??? How do we avoid overlap?
  + Is the rating system between GMaps (rounds to 1/10th star) vs Yelp (half-star) a problem?
    - No problem
    - We could learn a function from rougher estimates using regularization
* What do we do about small # of restaurants with low ratings
  + We don’t need a balanced dataset
* How do we get around small # of restaurants returned from API?
  + Get lots of restaurants (not necessarily lots of photos per restaurant)
    - At the very least in the 1000s
* How much focus do we have to put into architecture design?
  + This is problem-dependent
  + Use an appropriate architecture using prior work
  + Trial and error approach
  + Novel architecture *can* be introduced
  + Is this something we mention in technical methods?
    - Mention in interim report
* Classification vs regression??
  + More of a regression problem
    - Impact on diversity??

**WORK PLAN**

Problem Statement: Grant

* Rehash what we already have + add motivation

Data Gathering: Grant

* Discuss API and scraping methods
* Propose systematic way to use API

Prior Work: Hshmat

More Prior Work (combined):

Technical Methods: Vince