# Log Version 1

This is a continuition of the old log, which can be found [here](https://github.com/JinLi711/UIA/tree/master/Notes%20and%20Record%20Files).

# January 24, 2019

Brief write up on background and objectives. It would be a good idea to list all possible approaches that we want to try (and to sum up my write ups)

## Broad Objective:

Try to analyze the type of neighborhood of locations. Locations can refer both to a global sense(like cities or zip codes), or a local sense (like businesses)

One of the ways we are trying to do that is by looking at Yelp reviews, as we think that Yelp can provide a lot more information about the culture of the place than we think (as people usually just associate Yelp reviews with food).

## Background

Types of neighborhoods:

1. Traditional / Neighborly/ Local type

* the average place for average people to visit.

1. Corporate / Utilitarian type

* practical places.
* places where suits and ties meet.
* places where students study.
* think of like chains of cafes like Starbucks.

1. Bohemian type

* places where "rebels" (nonconformists hang out)
* supportive of diversity (ethnic, racial, gender, sexuality)

## Summary of My Research Before

I made a write up on word embedding before and pipeline for analysis, which can be found [here](https://github.com/JinLi711/UIA/tree/master/Notes%20and%20Record%20Files). (The file name is called Visual Representation Write Up).

## My more specific goals

What I'm doing is very different from what everyone else is doing. FAUI is more of a sociology group that uses qualitative analysis to understand demographics. I, on the other hand, an providing the quantatitive side. I've already done some of the analysis below in Possible Approaches to Analysis on Word.

So how can I analyze Yelp to provide meaningful insight to the types of neighborhoods?

### Possible Approaches to Analysis on Words

1. Given a predefined list of words, find the frequencies of those words in Yelp.

* Why this technique?
  + Using our intuition and experience, we can hypothesize that certain words that appear more often can correlate with the type of neighborhood.
* Pros:
  + easy to do
  + human judgement is used to decide the words.
* Cons:
  + we are not using the full capacity of the information in Yelp reviews.
  + there might be words on Yelp that we never thought of that can strongly correlate with the type of neighborhood.
  + Different words have the same meaning and a word can have multiple meanings

1. Build a clustering algorithm to cluster words that have similar meaning

* Why this technique?
  + To be able to visualize words on a scatterplot (I wrote more about this on the previous log)
* Pros:
  + we take care of the issue of different words having the same meaning

1. Create a graph/web connecting a word to other words, and finding the frequencies of each connection.

* Why this technique?
  + Take care of the possibility of missing frequencies of words that mean the same thing.
* Cons:
  + we're bounded by predefined words (since those predefined words act as the starting node). There's a possibility that the web may not touch words that strongly correlate with neighborhood types

### Questions I Want To Answer

1. How can I extract more meaning from Yelp reviews?

* Right now, I'm only looking at words by themselves, and I want to eventually be able to look at and understand words through groupings.

### Current Objective

1. Find a way to create a graph to connect starting words to words with closest meaning.

# January 25, 2019

1. Did some research on possible ways of finding synonyms.

* [Learning to rank](https://en.wikipedia.org/wiki/Learning_to_rank). Essentially a machine learning method for ranking which infomation is the most important or relevant.
  + Decided not to look into this because there are way better methods. This is here just in case I want to look into it again.
* Word2Vec Embedding
  + [Tensorflow](https://github.com/tensorflow/tensorflow/tree/r1.10/tensorflow/examples/tutorials/word2vec) example of Word2Vec
* [How to use machine learning to find synonyms](https://medium.com/@nikhilbd/how-to-use-machine-learning-to-find-synonyms-6380c0c6106b)
  + Candidate Generation
  + Generate all possibilities of synonyms
  + need acrynoym expansion (like CS to computer science)
  + use Word2Vec embedding
  + use historical user data (if a person searched shoes then searches sneakers afterwards, we can assume there might exist an association)
  + Use a pre-existing lexical synonyms. Ex. [WordNet](https://wordnet.princeton.edu/).
  + Synonym detection
  + use supervised learning
* [WordNet](https://wordnet.princeton.edu/) from Princeton.
  + Sort of used like a thesarus
  + Avaliable on Python NLTK
  + [Wordnet with NLTK](https://pythonprogramming.net/wordnet-nltk-tutorial/)
  + Not entirely extensive because
* [Sketch Engine](https://www.sketchengine.eu/price-list/).
  + For language exploration that ranges from translation to text analysis to creating lexicons. Does cost money but has a 30 day free trial.
* [Playing with word vectors](https://medium.com/swlh/playing-with-word-vectors-308ab2faa519).
  + Python code for Word2Vec and finding similarities. Will check out next week.
* [Get Busy with Word Embeddings – An Introduction](https://www.shanelynn.ie/get-busy-with-word-embeddings-introduction/). I know most of this, but quite a good article to reread
* I also [stared](https://github.com/JinLi711?tab=stars) a lot of repositories on NLP. Should check those out.

**Thoughts** After doing some preliminary research, I think it would be a good idea to take the set of words that Hyesun created, and run them through Stanford's GloVe. Then try to find closest words with the cosine similarity (since words are just vectors). Another good thing to do is run the list of words through NLTK's WordNet to find similar words. Will need to look into that. Then for each similar words, find the frequencies of those words.