# ORIE 4741 Project Proposal

Ryan Butler, Dae Won Kim, Peiyun Zhang

September 23, 2017

### 1. Background

Every year, Yelp releases a large volume of data the data source for the Yelp Data Challenge. The review system is one of the key functions provided by Yelp, which provides users with suggestions and reviews associated with a certain restaurant. After reading the reviews, users can also click three buttons to give feedback on a specific review. These three buttons are "Useful", "Funny" and "Cool". However, the review system currently is flawed because it is a "win-more" system where reviews with lots of positive reactions will be shown more prominently, causing them to get even more positive feedback. Meanwhile, other reviews that might be just as good often will not be seen by users because they never recieved the initial couple of positive reactions.

### 2. Objectives

We want to build a model to systematically determines which review might fit into a certain feedback category so that Yelp can build a better review recommendation system. Although this is not particularly useful for reviews that have already been "discovered" and have amassed lots of positive reactions, it will help Yelp to determine which "undiscovered" reeiews it should display to get them discovered by selecting reviews that are likely to be useful, funny or cool. This way, these perfectly funny, reviews won't have to rely purely on luck to get noticed and stand out from the majority of average reviews.

### 3. Proposed Methods

In order to develop training data, we will filter the already existing yelp dataset to find reviews that have exceeded a certain number of reactions in one of the categories. This will give us sufficient numbers of reviews that have been labeled by users as either funny cool or useful. The problem then is identifying "dull" reviews. The reviews that have not received large numbers of positive reactions cannot all be used because those could include reviews that aren't dull but just were unlucky and never became popular. Instead, all the authors of the reviews that had garnered enough positive reactions will be collected and then any review by those reviewers that had not received enough positive reactions will be considered dull. The logic is that if a reviewer has had success getting their reviews "discovered" in the past then they are likely to again, but not if their review is dull.

For each review, there are associated features such as the associated business, its location, average star ratings, as well as features associated features for the users. What we can also do is use methods like TF-IDF or word2vec to convert the review texts into numeric feature vectors. We can then use supervised learning techniques on these vectors such as ordinal regression methods or regression methods such as linear regression, KNN regressor, and CART. In order to increase the accuracy of our models, we may attempt to use bagging, boosting, and or stacking techniques depending on the relative performances of the weak learners we deploy. For exploratory analysis, we will be using clustering algorithms like k-means and Gaussian mixture models in order to understand the behavior of clusters and subgroups within the dataset. We anticipate that since these vectors may be high-dimensional, PCA could be used to lower dimensions to help visualize the data as well.

Finally, time permitting, we will use Recurrent Neural Networks to try to perform the prediction. We will generate word embeddings based on the most commonly used words, as well as have an "UNKNOWN" token to remove very uncommon words. Then, these word embeddings will be used as inputs to a stacked LSTM model. We will also try to use the Differentiable Neural Computer if we deem it computationally feasable and something we have time to implement. These models, as well as variations created by changing hyperparameters, will be compared with each other as well as to the conventional machine learning methods previously attempted.