Should I Invest it? Predicting Future Success of Yelp Restaurants

Xiaopeng Lu
University of North Carolina at Chapel Hill
Chapel Hill, NC, 27599
xiaopeng@live.unc.edu

Jiaming Qu
University of North Carolina at Chapel Hill
Chapel Hill, NC, 27599
jiaming@live.unc.edu

Yongxing Jiang
University of North Carolina at Chapel Hill
Chapel Hill, NC, 27599
yongxing@live.unc.edu

Yanbing Zhao
University of North Carolina at Chapel Hill
Chapel Hill, NC, 27599
yanbing@live.unc.edu

ABSTRACT

Online customer reviews are becoming more and more important in helping consumer make decisions. Therefore, it is also interesting to see whether consumer reviews can be used to predict future business success. In this paper, using yelp dataset from 2016, we aimed to predict if the restaurant will still open till 2017. We focused on multi-level feature selection and analyzed features that influence the most for the future success of restaurant. The balanced accuracy is 67.46%. The result shows that our text features failed to have significant indications for the future success of the restaurant, while non-text features, especially business features, do have strong correlation with future restaurant performance. Furthermore, we did error analysis on insignificant features and gave potential improvements.

CCS CONCEPTS

• Information systems~Wrappers (data mining) • Computing methodologies~Information extraction

KEYWORDS

text mining, advanced research computing, machine learning, natural language processing

ACM Reference format:

Xiaopeng Lu, Jiaming Qu, Yongxing Jiang and Yanbing Zhao. 2018. Should I Invest it? Predicting Future Success of Yelp Restaurants. In *PEARC '18: Practice and Experience in Advanced Research Computing, July 22-26, 2018, Pittsburgh*, PA, USA. ACM, New York, NY, USA, 6 pages. https://doi.org/10.1145/3219104.3229287

PEARC '18, July 22–26, 2018, Pittsburgh, PA, USA

ACM 978-1-4503-6446-1/18/07…$15.00
https://doi.org/10.1145/3219104.3229287

1 INTRODUCTION

Consumers often refer to online reviews before making decisions. Aiming to make online reviews play a greater role, we not only want to let these reviews serve individual consumers as references, but also want them to be useful to help investors make business decisions. Therefore, according to the review dataset of restaurants, we extracted and analyzed the implicit and explicit features of those reviews, then trained and built a model to predict whether a restaurant will be still open or shut down after a period of time.

2 RELATED WORK

In the recent years, there have been many studies on e-commerce websites, most of which focus on semantic and sentiment analysis. The ultimate goal of these studies is to extract individual consumers' most concerned aspects and forecast their emotions and decisions.

Prior work in this field mainly studied what kind of information can be retrieved from the reviews. Some also tried to find which restaurant features have the most impact on the restaurant's success and predict the rating and success of the restaurant.

As a modern and information-rich dataset, the Yelp dataset has become the main resource for predicting the success and failure of restaurants. For example, Feng, Kitade and Ritter [1] considered the successful restaurant has more than 37 reviews and at least 3.5 stars on Yelp. They used various prediction models from neural network, logistic regression to SVM and determined that their neural network was the best model. They concluded that restaurants with incomplete features are generally less successful.

Another study by Camillo, Connolly and Kim [2] attempted to focus on the independent restaurant operators in the San Francisco Bay Area. The study found the emotional factors and operational factors related to the restaurant success. The study concluded that successful restaurants often have a clear, well-crafted business plan, and are distinct in the marketplace. Moreover, most of these restaurants have strong employees and located in areas with sufficient demand.

The project conducted by Kong, Nguyen and Xu [3] aimed to identify the key features people in different countries look for in their dining experience, in order to predict international restaurant success. They used methods such as Naive Bayes, SVM, decision trees and natural language processing methods to identify the most informative features from restaurant review texts.

In the research completed by Wang, Zeng and Zhang [4], it aimed to predict new restaurant success and ratings. Wang utilized a variety of binary and multi-class classification algorithms and did sentiment analysis on restaurant reviews as well to improve accuracy. The conclusion was that the sentiment analysis increased the accuracy of classification by 32%, demonstrating that food quality and other factors within user's review text definitely affect a restaurant's average rating.

3 DATA COLLECTION

In this section, we provide a general description of data selection and formulation for the following feature engineering and experiment.

Yelp dataset description. We got the data from Yelp website [5]. Yelp updates its open dataset regularly for its Dataset Challenge. In order to analyze the success or failure of a restaurant in a long period of time, we retrieved two pieces of datasets with same formats but different release dates. One was released in 2016 and the other in 2017.

Each piece of dataset includs six JSON files. In this paper, we only used two *business.json* (released in 2016 and 2017) files and two *review.json* (released in 2016 and 2017) files. The *business.json* file contains businesses from selected cities, where each business has 15 attributes. The attributes used in this paper are *"business_id"*, *"name"*, *"city"*, *"state"*, *"latitude"*, *"longitude"*, *"stars"*, *"review_count"*, *"categories"*, and *"is_open"*. Most of these attributes are self-explanatory. *"is_open"* is the attribute used to annotate whether the business is open or not. Those which were open till the dataset release date were marked as *1*, and those which were closed were marked as *0*.

The *review.json* file contains all user reviews till the dataset release date. It has 9 attributes, and in this paper we used *"review_id"*, *"user_id"*, *"business_id"*, *"stars"*, *"date"*, and *"text"*. Review and business datasets could be connected on *"business_id"*, which means the *review.json* file stores all the reviews for all the business recorded in the *business.json* file.

Due to the large size of released json files (about 3.7 GB for each review.json file), we utilized Google cloud platform to parse and did basic data cleaning (removing html tags and blanks). We saved the parsed objects into tab delimited file for future use, and marked them as business_2016, business_2017, review_2016, and review_2017.

Dataset formation. In this paper, we aimed to use information collected at 2016 to predict whether a restaurant is still open or not till 2017. In order to do so, we first removed all closed businesses in the *business_2016* dataset to make sure all businesses in the dataset were still open till 2016. After that, we selected businesses with keyword *"restaurant"* in the *"category"* attribute and dropped the rest. Till this point, we got all the restaurants which were open till 2016 and saved this collection as *open_restaurant_2016*.

The next step was to check whether these restaurants were still open or not in 2017. Because Yelp regenerated the *business_id* value in the dataset at every round, we were unable to use the *business_id* to get corresponding restaurants in the *business_2017* dataset directly. Instead, we used some matching criteria to solve this problem. For both *open_restaurant_2016* and *business_2017* datasets, we concatenated *"name"*, *"city"*, *"latitude"* and *"longitude"* attributes, then formed a new attribute called *"identify"*. For both *latitude* and *longitude* attributes, we dropped last 3 decimal places and only kept the first 3, for we observed that in some cases, the same restaurant showed slightly different latitudes and longitudes in different datasets. We did not use "*address*" in matching criteria because the form of this attribute was different in two datasets. Below is a detailed example for illustration (see in Figure 1). The top one is from *open_restaurant_2016* and the other from *business_2017*. They are matched as the same restaurant though *latitude* is different.

business_id	name	address	city	state	latitude	longitude	stars	review_count	categories	is_open	identify
6iIJq_05xRgek_8qUp36-g	Steak 'n Shake	650 E Waterfront Dr Homestead Munhall, PA 15120	Munhall	PA	40.413496	-79.904456	2.0	60	Burgers;Fast Food;Restaurants	1	(steak 'n shake, munhall, 40.413, -79.904)

business_id	name	address	city	state	latitude	longitude	stars	review_count	is_open	categories	identify
nxAo0oDgsNlduzUuuUL4Fg	Steak 'n Shake	650 E Waterfront Dr E	Munhall	PA	40.413445	-79.904447	2.0	79	1	['Steakhouses', 'American (New)', 'Restaurants...	(steak 'n shake, munhall, 40.413, -79.904)

Figure 1: Illustration of matching criteria.

There were 20124 restaurants open in 2016, and 15784 out of them had a match in the business_2017 dataset. 1047 out of these 15784 matched restaurants were closed down after one-year period, i.e. 2017. In order to make the final dataset evenly split, we randomly sampled 1047 from restaurants that were still open till 2017. The final dataset contained 2094 restaurants, which was formed by combining 1047 restaurants that were closed till 2017, and 1047 restaurants that were still open till 2017. The figure below shows the whole process of dataset formation (Figure 2).

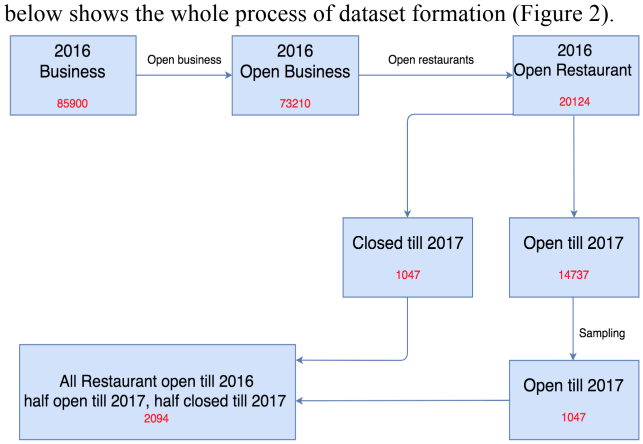

Figure 2: Dataset formation process.

4 FEATURES

We grouped our features into two top categories: text feature and non-text feature. In text feature category, we built both Unigram and Bigram as two feature groups. In non-text feature category, we marked three middle-levels: Trend, Business, and Location. Under each middle level feature group, there were several bottom-level features. Below is a summary of these features (see in Figure 3).

Text Features	Unigram	Good
		Bad
	Bigram	Sanitation
		Location
		Service
		Taste
Non-text Features	Trend	Review Star Loss
	Business	Review Count
		Chain Restaurant
		Return Guest Count
		Restaurant Type
	Location	Nearby Restaurant Comparison
		City Economic Status

Figure 3: Level of features and categorizations.

Text Features (10). We created this group of features in order to analyze if there are any specific factors that might be strongly correlated with the development of restaurant. As a result, instead of setting word occurrence threshold to get unigram, we generated domain-specific-keywords dictionary and counted word occurrence as features. Before generating text features, we grouped all reviews by "*business_id*" and concatenated same-restaurant reviews into one field. 126,715 pieces of reviews were grouped to 2087 restaurants. Then instead of analyzing review piecewise, we regarded all reviews under the same restaurant as a whole and analyzed it holistically.

Our initial strategy was to imagine some domain specific words and count occurrence and co-occurrence in the review. The result was not promising because the feature value turned out to be too sparse, which means that our key-words dictionary were too few to catch important information. As a result, adding more words was a potential solution. In order to get context related similar words, using Google cloud platform, we used Word2Vec[9] and trained a model on the reviews of all open restaurants in 2016 with 10% test set excluded. After that, we enriched our dictionary by a significant amount. More details about this part will be provided in the discussion section.

- Unigram features (2). Two features were designed in this category: *"unigram_good"* and *"unigram_bad"*.
    - *"unigram_good"*: we calculated frequency of positive words, such as "love", "nice", "delicious", "favorite", etc.
    - *"unigram_bad"*: we included words such as "nasty", "noisy", "disappointed", "cockroach", "fly", "mosquito", etc.
- Bigram features (8). All words pairs in our bigram feature category are in the form of noun + adjective. Four specific domains were designed: sanitation, location, service, and taste. Each domain contains a pair of features: domain_good and domain_bad. We caught word pair sentence-wise, which means that in given sentence, if word pair occurs (anywhere in the sentence, does not have to be together), we counted its occurrence.
    - Sanitation_good (bad): We designed specific word pairs with indication in sanitation field, such as "environment...clean" and "atmosphere...quiet" for good sanitation, while "environment...nasty" and "table...dirty" for bad sanitation.
    - Location_good(bad): We used word pairs such as "place...cool" and "parking...easy" as the indication of good location, while "place...crowded" and "bar...boring" for bad location.
    - Service_good(bad): We found word pairs such as "waiter...helpful" and "service...fantastic" for indicating good service, while "waitress...worst" and "staff...disrespect" for bad service.
    - Taste_good(bad): We used word pairs such as "drink...best" and "dessert...wonderful" for indicating good taste, while word pairs like "food...nasty" and "appetizer...disgusting" for bad taste.

Non-text Features (7). Non-text features were also included in the study. Unlike text features which were generated from review texts, non-text features were more like statistical data generated from attributes in the dataset. We wanted to use non-text features to describe a restaurant in a more objective way. We identified three middle-level features for the non-text features: Trend, Business and Location. Each of them have lower-level features.

- Trend (1).
    - Review Star Loss (1). If the review star of a restaurant had a trend of decreasing, it would probably be closed down in the future. Otherwise, if it had a trend of increasing, it might be more likely to keep open. For each restaurant, we fitted a linear regression between the review star (Y) and the date(x):

      $Y = ax + b$

For certain restaurant, we marked the date of the first review as x=0, then the x value for every other piece of reviews is the number of days passed since this first review date. Then we extracted the coefficient $\boldsymbol{a}$ as a feature, which indicates the trend of the review star in a given period of time.

- Business (7).

  - Review Count (1). It was hypothesized that restaurants with more review counts would be more popular and have more guests, which could make it less possible to be shut down. We calculated the total review count for each restaurant and used the number as a feature.
  - Chain Restaurant (1). Chain restaurant might have more stable business style and more economic backup, which could make it stand the test of the market at the initial stage and unlikely to be closed down. We identified restaurants which appeared three times or more in the dataset as chain restaurants and marked it as 1, the rest were marked as 0.
  - Return Guest Count (1). It was hypothesized that a higher return guest amount means stronger user loyalty, i.e. more people liked it and were willing to come back, which might largely reduce the possibility of being closed down. In the review dataset, we identified the return guest for a restaurant when a single customer had two or more reviews for a particular restaurant. Then we counted the number of return guest for each restaurant.
  - Restaurant Type (4). It was hypothesized that some kinds of restaurants would be more welcomed and restaurants with more characteristics would be more popular than others. As a result, we extracted restaurant types from "category" field, and constructed four features: "thai food", "american food", "mexican food", and "fast food". If certain restaurant had these keywords in its "category" field, the corresponding feature value would become 1 instead of initial value 0.

- Location (4).
  - Nearby Restaurant Comparison (3). As we analyzed the distribution between review stars and close/open rate, we found that there was no strong correlation between review star and restaurant open. In other words, compared with restaurants with high stars, those with relatively low review stars are not necessarily more likely to shut down. So, we utilized a new way to measure restaurant performance: we intended to compare the star of each restaurant with the mean star of its nearby restaurants. Practically, we chose radius 1, 2, and 3 miles as the radius of circle, and generated 3 features to see which range might give the best predictive power.
  - City Economic Status (1). The economic status of certain city might affect the business situations for all restaurants in it. For example, if the economy at New York crashes in 2016, there might be more restaurants shut down during this period, compared with regular time. Therefore, we collected the 2016 Per Capita Income data for each city and normalized it as a feature.

5 EVALUATION METHODOLOGY

Since it is a binary classification problem, we selected logistic regression as our classifier and trained a model based on features provided above.

We trained and evaluated the model using 90/10 train/test split. Because some of our features needed to be generated through reviews, we chose to first cut the dataset into training and test sets, then to do feature engineering. More details will be provided in the discussion section. We presented accuracy and precision as our evaluation metrics.

Feature ablation study was also done in order to analyze how each group of features contributed to the overall performance. In order to do it, we removed one feature group each time (with replacement) and built the model with remaining features, then tested it using 90/10 train/test split method as we described above. Say, if in total there were n feature groups, we built model n times, each time using features from (n-1) feature groups by removing different feature groups one at a time. The extensive feature ablation table was shown in the next section.

6 RESULTS

The final overall accuracy is 67.46%. Since our dataset is balanced and the baseline accuracy is 50%, 67.46% appears to be an acceptable number. Also, since our goal is to help people decide if certain restaurant is worth investing, precision for the "open" class is critical. Our precision rate for "open" class is 73%, which means that in all restaurants we tried to predict, 73% restaurants which were predicted as open turned out to still open after a one-year period. In order to better understand how each feature groups contributed to the performance, we presented feature ablation study table below. The number below each feature group represents the accuracy after this group being removed. The number inside of bracket corresponds to the percentage drop compared with the overall accuracy (67.46%).

-Text Features 0.6650(-0.96%)	-Unigram 0.6698(-0.48%)	-Good 0.6746
		-Bad 0.6746
	-Bigram 0.6794(+0.48%)	-Sanitation 0.6698(-0.48%)
		-Location 0.6698(-0.48%)
		-Service 0.6650(-0.96%)
		-Taste 0.6794(+0.48%)
-Non-text Features 0.5502(-12.44%)	-Trend 0.6698(-0.48%)	-Review Star Loss 0.6698(-0.48%)
	-Business 0.5885(-8.61%)	-Review Count 0.6842(+0.96%)
		-Chain Restaurant 0.6172(-5.74%)
		-Return Guest Count 0.6746
		-Restaurant Type 0.6698(-0.48%)
	-Location 0.6555(-1.91%)	Nearby Restaurant Comparison 0.6650(-0.96%)
		City Economic Status 0.6650(-0.96%)

Figure 4: Feature ablation study.

As we can see from above, non-text features contributed most of the accuracy. And inside of non-text features, business feature groups are the most important one. This means that feature groups under business category are significant for predicting the open of restaurants.

We further checked how each feature in the business category performed, to understand which features are the most important for prediction. We found "chain_restaurant" did the most contribution: after removing it, the accuracy dropped by 5.7%. And actually, "review_count" feature hurt the prediction accuracy by 1%.

Text features did not contribute as much as we expected. The bigram feature groups even hurt the performance. We checked how each feature in this category related to the accuracy, and found that bi_taste features actually hurt the overall accuracy. Other three pairs of bigram features, though did not hurt the performance, failed to contribute significantly in this model.

7 DISCUSSION

The final test accuracy with our feature engineering and logistic regression classifier is 67.46%. Michail Alifierakis did similar project before, and got the result at around 80% [10]. However, that accuracy was based on unbalanced dataset, and the baseline accuracy is around 77%. As a result, it is reasonable to say that our model outperformed the model generated by Michali before. Moreover, the dataset Michali constructed was the yelp data at 2013 and 2017, which is a 4-year period. While in this paper, our dataset was constructed by using 2016 and 2017 yelp dataset. Compared with 4-year period dataset, 1-year period might be harder to predict: taken our trend feature for example: one-year period might be too short to see how the trend fluctuated. Also, city economic status might not instantly had its influence on the business: it requires some time to reflect the economic status onto individual restaurants. Therefore, it is worthy of trying to collect a longer-time-period dataset and test our model performance in the future.

Text feature group did not contribute that much to the overall accuracy, while it does have something that we can analyze and learn from. First of all, removing **unigram** text features did not affect the model accuracy at all. This implies that general sentiment did no help in predicting the open of the restaurant. This conclusion makes no surprise, since it is consistent with the concept we mentioned before: restaurants with low review stars were not necessarily more likely to shut down.

Also, looking at **bigram** feature statistics, we can find that although three of four bigram feature groups did help to the model performance, when removing all bigram features, the model even become better. Though this result seems inconsistent, it might imply that these features have strong correlations with each other: each one had its own significance, while if throwing them all into the model, the accuracy dropped down. Trying to find how and why this happened, we went back to text feature selection phase. At first, we constructed our own domain specific dictionaries just by imagination, and it did not perform well. Therefore, we looked back to the distribution of these features. The distribution table is presented below. As we can see, all these features turned out to be sparse: most of them were all zeros. As a result, it is necessary to find a way to enrich our dictionaries and let our dictionaries catch as many domain-specific information as possible.

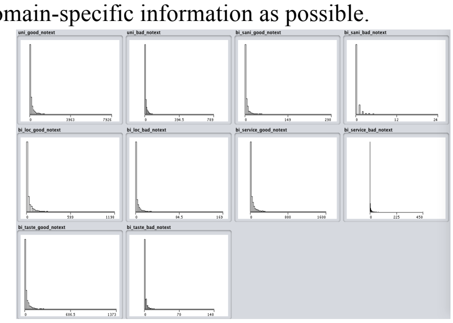

Figure 5: Text feature distribution analysis.

In order to enrich our dictionary and better catch domain specific expressions, we used gensim package in python to train a Word2Vec model in order to find "contextually" similar words from the corpus [9].The corpus contains all open restaurant reviews in 2016, excluding 10% testing set which we already split beforehand. After training this model, we fed in the most representative words $w_{1\ldots n}$ for each domain (e.g., "hygiene" for sanitation field; "waiter" for service field) with query "most_similar(w, topn=20)". After that, the model returned 20 top correlated words based on this corpus. Then, we handpicked noun or adjective as we wanted, and enriched our domain-based dictionaries.

As we reviewed the selected words, one potential drawback of this method is that as words increase, it becomes much harder to distinguish and tell the difference among groups. For instance, we added "window" as one keyword in sanitation field, wishing to catch "window...clean", "window...dirty", etc. After using Word2Vec, one similar word for "window" is "place". Therefore, we then added "place" into our sanitation-dictionary. However, "place" was already one of the keywords in location-dictionary. As the number of words increased, our domain specific dictionary became vague in boundaries and turned out to be interrelated with each other. Finally, these bigram features might caught keywords much more than we expected and it turned out to be a bad idea.

One potential solution could be to select words more strictly, and drop words which are ambiguous. Another solution might be to fine tune the hyper parameter of Word2vec model. "window_size" is a hyper parameter which controls how many context information should be stored. Similar to choosing k for k-Nearest-Neighbor classifier, playing with window size could help construct the model we want [8].

Business feature group contributed the most accuracy for the whole model. Looking into this group, we found that ***"chain_restaurant"*** feature had the strongest correlation with business open. This result is the same as Michali's work [7], which shown that a chain restaurant was less likely to close compared to normal restaurant. In the same group, however, other three features did not perform that good compared with "*chain_restaurant*" feature. We generated ***"review_count"*** features to try to measure the popularity of the restaurant, and see if the popularity of the restaurant affects its future success. And it turned out to be not true: the accuracy increased when this feature being removed. Some aspects might be overlooked when we generated this feature: we did not normalize the review count by the number of days since the first review. Think about this: a 3-month-old restaurant having 1000 reviews is totally different from a 10-year-old restaurant having the same number of reviews. As a result, the overlook of normalization might be a potential reason why this feature did not work. We generated ***"return_guest_count"*** feature as well and thought that a restaurant might be excellent if certain person come there several times and leave multiple reviews. However, experiment show no correlation between this feature and the restaurant open. When we stepped back and looked into those with high return guest count, we found that our hypothesis is not necessarily true: some restaurants with very high "*return_guest_count*" actually closed. Further looking into these circumstances, we found that some second reviews (same person's second review for the same restaurant) were more negative than the first one. In this case, return guest review worked against our hypothesis. We underestimated the complexity of review contents and just counted the number of return guest, and this might be the reason why this feature did not work.

Location feature group also appears to have some significance. "***Nearby_restaurant_comparison***", though did not greatly improve the accuracy, still shew some influence on it. This feature implies that restaurants do have to care about the "peer pressure" and try to do better than average in certain area. "***city_economic_status***" also worked: instead of releasing all the data, Yelp only released data from nine metropolitan cities for analytic usage. As a result, it becomes difficult to study if city economic status would affect business closure. One potential improvement with current dataset might be, instead of collecting Consumer price index for only 2016, we could collect the same type of data for previous few years, and see how trend goes for each city. This trend could potentially be a new feature.

8 CONCLUSION

As we stated before, the goal of this paper is to look for features in order to better predict the future success of restaurant. It turns out that our model works fairly well, and achieved a 67.46% accuracy in the end.

We generated our features through several aspects, and both text and non-text features were covered. Non-text features are more important in the model. Chain restaurant feature turned out to be the most significant one, and other features, such as trends, nearby comparison, and economic status all have their own influence on the whole model. However, the performance of text features was not so good as expected because word patterns were not analyzed effectively. As our text features do have some prediction power, it would be interesting and promising to do more work on this side and see how deep we can mine into the review to find the correlation between business success and user reviews.

REFERENCES

[1] Feng, Jason., Kitade, Naho., & Ritter, Matthew. (2015). *Determining Restaurant Success or Failure.* Retrieved from http://www.cs.dartmouth.edu/~lorenzo/teaching/cs174/Archive/Winter2015/Projects/finals/fkr.pdf

[2] Camillo, A. A., Connolly, Daniel J., & Kim, W. G. (2008). Success and Failure in Northern California. *Cornell Hospitality Quarterly*, 49(4).

[3] Kong, Angela., Nguyen, Vivian., & Xu, Catherina. (2016). *Predicting International Restaurant Success with Yelp.* Retrieved from http://cs229.stanford.edu/proj2016spr/report/062.pdf

[4] Wang, Aileen., Zeng, William., & Zhang, Jessica. (2016). *Predicting New Restaurant Success and Rating with Yelp.* Retrieved from https://web.stanford.edu/class/cs221/2017/restricted/p-final/wizeng/final.pdf

[5] Yelp Dataset. (2016, 2017). Retrieved from https://www.yelp.com/dataset

[6] Mikolov, T., Sutskever, I., Chen, K., Corrado, G., & Dean, J. (2013). Distributed Representations of Words and Phrases and their Compositionality. *NIPS'13 Proceedings of the 26th International Conference on Neural Information Processing Systems,*3111-3119.

[7] Alifierakis, M. (2018). Using Yelp Data to Predict Restaurant Closure – Towards Data Science. Retrieved from https://towardsdatascience.com/using-yelp-data-to-predict-restaurant-closure-8aafa4f72ad6

[8] McCormick, C. (2016). Word2Vec Tutorial - The Skip-Gram Model. Retrieved from http://mccormickml.com/2016/04/19/word2vec-tutorial-the-skip-gram-model/