Recommendations for Restaurant Improvement using Random Walk Based Opinion Mining

Ekta Rita
University of Southern California
erita@usc.edu

Shreya Venkatesh
University of Southern California
shreyave@usc.edu

Srisha Raviillu
University of Southern California
raviillu@usc.edu

ABSTRACT

Some restaurants perform better than others despite having very similar population geography and service attributes. In this paper, we aim to single out the probable causes that may contribute to the inferior performance of a restaurant and in turn recommend improvements that lead to gaining competitive edge. Sentiment analysis using Random Walk is used to assign polarities to the words in common in the combination of reviews and tips of the two restaurants to understand how an attribute or a feature contributes to the restaurant's performance. Accuracy is used as the evaluation metrics, and word cloud is used for visualization.

Categories and Subject Descriptors

H.3 [**Information Storage and Retrieval**]: Information Search and Retrieval; G.2 [**Discrete Mathematics**]: Graph Theory

Keywords

Opinion Mining, Random Walk, Page Rank, Jaccard Similarity, Yelp Dataset Challenge

1. INTRODUCTION

Ample research has been done on recommender systems (or a recommendation systems/platforms/engines) in attempt to predict preference or rating. Existing recommendation systems revolve around personalizing user experience or enabling targeted marketing, they suggest users to products based on user similarity, or products to users based on product similarity. Specific to restaurants, recommender systems suggest restaurants to users. This benefits the users by suggesting them services they are likely to enjoy, it benefits restaurants by widening their customer base. This paper differs in the sense that it compares a restaurant with its competitors and recommends upgrades, refinements, or renovations that will directly contribute to meeting - if not outperforming - its competitors' standards.

Yelp data set chosen for implementation of this project provides reviews and tips for every business. Ideally, all reviews and tips could potentially help the restaurant, but most businesses have limited resources and need to prioritize the improvements that might benefit it the most. It would be right to assume that restaurants - barring the ones that primarily serve tourists - cater to the crowd within a given radius. Based on the aforementioned assumption, this paper compares restaurants in the same neighborhood in order to ensure the mere choice of location has no bearing on its performance.

There could be numerous reasons why a restaurant is not doing well, a primary reason can be lack of appetite for the cuisine served by the restaurant in its locality. This paper compares restaurants serving the same cuisine to provide accurate recommendations for improvement. There will be recommendations based on the menu, but that will mostly pertain to the quality of what is already being served.

Restaurants serve a wide range of customers, comparing restaurants with high attribute-similarity will ensure a restaurant's performance is measured against another restaurant with same target demographic, that is its closest competitor.

For the purpose of actual refinement recommendations we intend to extract important information from user reviews of restaurants. For this we require assignment of sentiment orientation to important words in the review. SO-PMI [4] (Sentiment Orientation with Point-wise Mutual Information) achieves assignment of scores to words based on a word corpus. However, contextual information would be missed by using the above mentioned approach. For assigning sentiment scores to words in context we use random walk. The approach of using random walk for sentiment orientation has been tried with successful results in Baccianella et al. [1]. Sentiwordnet 3.0 uses semi-supervised random walk with synset based interconnections to assign scores to words on the wordnet [2]. Similar work for sentiment classification of documents has been proposed by Mingzhi et al. [3] using SO-PMI [4] weights for edges.

For satisfying our requirement we combine the methodologies used in [1] and [3] to derive an efficient graph representation that uses a semi supervised random walk model to assign sentiment scores to induvidual words of the review while preserving contextual information with high efficiency.

The rest of this paper is organized as follows: Section 2 describes our experiment and all work-flows in detail. In Section 3, we discuss and present our results. Finally we sum up our contributions in the paper in Section 4

2. EXPERIMENTAL SETUP

Our proposed solution consists of 4 phases as illustrated by Figure 1. The first phase involves data extraction, followed by data selection and similarity matching to find localized

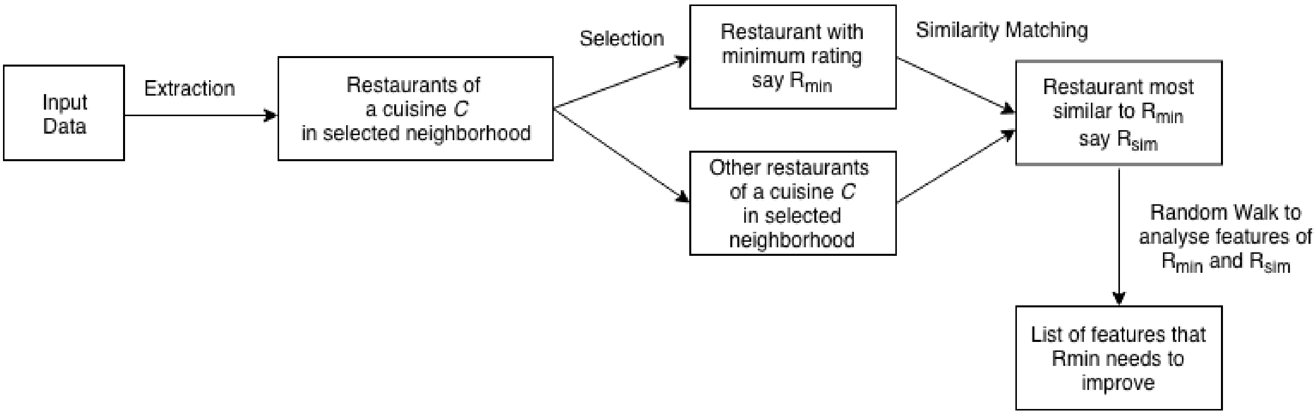
Input Data
Extraction
Restaurants of a cuisine C in selected neighborhood
Selection
Restaurant with minimum rating say R_{min}
Other restaurants of a cuisine C in selected neighborhood
Similarity Matching
Restaurant most similar to R_{min} say R_{sim}
Random Walk to analyse features of R_{min} and R_{sim}
List of features that Rmin needs to improve

Figure 1: Improvement Extraction Framework

similar restaurants. Finally random walk is performed to recommend improvements to the restaurant that has a lower performance rating.

The data set we have used is obtained from the the Yelp's Dataset Challenge and in particular, the following data is of importance to us:

- Business: Details about the business - (business_id, neighborhood, city, stars, review_count, attributes, categories)
- Review: Reviews the business has got - (business_id, stars, text)
- Tip: Any tips the business has received - (text, business_id)

2.1 Extraction

The Yelp data is converted to CSV format and then filtered out to extract all the restaurants from the business data by selecting only those records that have the words ”restaurant” or ”food” in categories or attributes column. After this elimination process we obtain a data set of 61k restaurants from approximately 162k businesses.

The extracted restaurant data is grouped by city and neighborhoods and one neighborhood from each city with the most data is selected from each of the four different cities. From the so obtained data, the data is split according to the cuisine $\boldsymbol{C}$. On analysis of the cuisines we found the that the most common cuisines or restaurant types in the data set included Bars, American, Sandwiches and so on. The top ten most common cuisines found in the data set are illustrated in Figure 2.

From analysis we decided to chose Bars, Pizzas, Chinese and Italian as our cuisines/ categories of interests. To ensure data availability for further processing, only those restaurants from each cuisine which had at least 50 reviews were extracted.

2.2 Selection

The next task was to select the worst performing restaurant $\mathbf{R}_{min}$. In this step, we selected the restaurant that had lowest ratings with a high number of reviews. A high review count is necessary as this indicates that many users who have reviewed the restaurant have rated it low. The remaining restaurants are used to find the similarity with the worst performing restaurant.

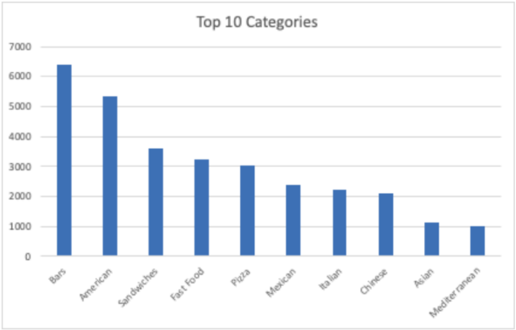

Figure 2: Top 10 Categories

2.3 Similarity Matching

Similarity matching is a crucial step in our work-flow. We use the Jaccard Coefficient to determine the similarity between the worst performing restaurant $\mathbf{R}_{min}$ and the other restaurants in the neighborhood with cuisine $\boldsymbol{C}$ as follows:

$$\text{J(Worst,Other)}=\frac{\text{Intersection}(\text{Worst}_{Attribute},\text{Other}_{Attribute})}{\text{Union}(\text{Worst}_{Attribute},\text{Other}_{Attribute})}$$

Here attributes represents the boolean attributes obtained from the ”attributes” field in the business dataset. From the similarity matching we obtain $\mathbf{R}_{sim}$, which is the most similar restaurant to $\mathbf{R}_{min}$, but has a better rating. Also, the difference between the rating magnitudes of $\mathbf{R}_{sim}$ and $\mathbf{R}_{min}$ is atleast 1.

2.4 Semi-Supervised Random Walk in Context

A random walk in a graph environment is considered as iterative process that starts with a random node and at each step, it follows a path of random vertices via outgoing edges. We use Page Rank, which is a type of random walk which follows a path to another node with probability p. The equation for page rank is given below :

$$PR(A) = (1-d)+d(PR(T1)/C(T1)+...+PR(Tn)/C(Tn))$$

where A is a node in the graph, T1, T2..Tn are outgoing edges from A and C(T1), C(T2)..C(Tn) are the probabilities of jumping from A to T1, T2..Tn respectively. d is the dampening factor.

To build a graph representation of the data, for $\mathbf{R}_{sim}$ and $\mathbf{R}_{min}$, we extract the reviews and tips from the datasets. Since we do not differentiate between reviews and tips, the reviews and tips are extracted and combined into a single data set. Certain preprocessing steps such as stop-words removal and lower case transformations were performed during the extraction process. For this step, the data extracted for each of $\mathbf{R}_{sim}$ and $\mathbf{R}_{min}$ are treated as separate entities and go through the same treatment that follows.

Since our method uses a semi-supervised model, we used the help of object scores, positive scores and negative scores found in Sentiwordnet 3.0 [1] to construct a positive label dictionary, $\mathbf{L}_{pos}$ and a negative label dictionary, $\mathbf{L}_{neg}$, by combining all the reviews for a restaurant as a single text entity and extracting words based on the following rules. For each word in the review, the following are applied :

- If word in Sentiwornet and obj_score $<$ 0.8 and pos_score $>$ 0.6 - add (word, pos_score) to $\mathbf{L}_{pos}$

- If word in Sentiwornet and obj_score $<$ 0.8 and neg_score $>$ 0.6 - add (word, neg_score) to $\mathbf{L}_{neg}$

We build two graphs for each restaurant, $\mathbf{G}_{pos}$ and $\mathbf{G}_{neg}$ with almost a similar structure, for each of $\mathbf{G}_{pos}$ and $\mathbf{G}_{neg}$, the vertices of the graph form the words in the reviews and tips and two vertices are connected by an edge if they co-occur in the same sentence. The weights of each of the edges are Markov probabilities given by the equation below :

$$p_{ij} = Pr\{X_n = j|X_{n-1} = i\}$$

Where p$_{ij}$ is the weight of the edge and i and j are words that co-occur in a sentence. For $\mathbf{G}_{pos}$, we set words in $\mathbf{L}_{pos}$ with their sentiwordnet pos_score as the personalizing parameter for node weights. Likewise, for G$_{neg}$, we set the words in $\mathbf{L}_{neg}$ with their neg_scores as personalizing parameter for node weights.

We used NetworkX library to perform Page Rank on the constructed graphs using 0.85 as the dampening factor and 200 maximum iterations. The final sentiment orientation of a word is Positive if it has a higher rank in $\mathbf{G}_{pos}$ and negative if it has a higher rank in G$_{neg}$.

3. RESULTS

Once the sentiment orientation scores for both $\mathbf{R}_{min}$ and $\mathbf{R}_{sim}$ are obtained, the features of these 2 restaurants are compared and only those features are considered which are common to both. After filtering the unwanted features (words that do not correspond to object words) using manually created word lists, only those features are used whose

- negative rank $<$ positive rank for $\mathbf{R}_{sim}$ (feature has a good impact in $\mathbf{R}_{sim}$) and

- negative rank $>$ positive rank for $\mathbf{R}_{min}$ (feature has a bad impact in the worse performing restaurant $\mathbf{R}_{min}$)

This is the final features list which indicates what features needs to be improved in $\mathbf{R}_{min}$.

To observe the performance of our algorithm we have used the accuracy measure. The accuracy is calculated by comparing the sentiment score for a word in both $\mathbf{R}_{sim}$ and $\mathbf{R}_{min}$ by iterating through (stars, reviews) pairs to check if the word is associated with good (>=3) rating or bad (=<3) rating. Table 1 contains figures which represent the accuracy measures that were achieved after performing the Random-Walk algorithm on $\mathbf{R}_{min}$ and $\mathbf{R}_{sim}$ on 4 different neighborhoods with the local most popular cuisine.

City-Neighborhood-Category	Accuracy
Montreal-Ville Marie-Pizza	72.22
Cleveland-Goodrich Kirtland-Chinese	71.84
Missisuaga-East Credit-Bars	71.42
Pittsburg-Downtown-Italian	65.59

Table 1: Accuracy Measure

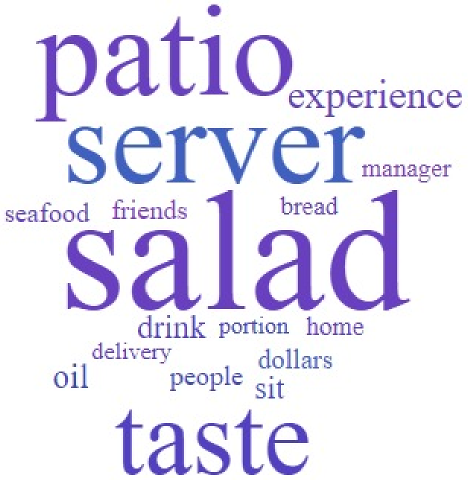

Figure 3: Recommended Features for Improvement

Figure 3 represent a wordcloud of the features that require improvement in the worst rated pizza place of the best neighborhood at Montreal, where the density of the feature indicates its importance in increasing the rating.

Some of the reviews for the feature "Patio" in the worst restaurant for Montreal-Pizza is as follows:

- ”There was only one server handling like the **patio** that had 4 tables filled with customers. By the time we got there all the other tables have already ate i assume and were just having beer or coffee...”

- ”Tables were crowded, unkept and unorganized in the front **patio**. Was seated immediately by the staff. Given water menus and then told he’d return to take our drinks order. In return of our order we received luke warm beer from the taps (goose brand beer) and a slightly cool shock top...”

Some of the reviews for the feature ”Patio” in the best restaurant for Montreal-Pizza is as follows:

- ”Great sit down pizza place on the Latin Quarter. Plenty of seating inside as well as in the outdoor **patio**.”

- ”My girlfriend and I came here and ordered some pizza. Service was great as was the food. Highly recommend. Nice area, quiet, and with a lovely **patio**.”

From the above reviews, we understand that the feature ”Patio” is mentioned negatively in the worst restaurant and positively in the best restaurant. Thus, it needs to be improved in the worst restaurant to get good ratings.

4. CONCLUSION AND FUTURE WORK

This paper compares and analyses the least rated restaurant’s business with its competitors to provide reliable data on aspects whose improvement could directly affect the rating and in turn improve its performance, but the same metrics can be used by any restaurant to gain competitive edge. The sentiment scores of the words obtained after convergence in Random Walk can help the restaurants know the aspects affecting its performance.

The approach followed in this paper for restaurants can be scaled to include other businesses as well. Using latitude and longitude may be a better way to select businesses serving the same demographic rather than using the value mentioned in neighborhood field. Considering ratings and reviews from a recent time period can help track performance of the restaurant and compare it with its past performance and its competitors, this eliminates the problem of using reviews that might not be relevant any longer.

5. REFERENCES

[1] Baccianella, Stefano, Andrea Esuli, and Fabrizio Sebastiani. ”Sentiwordnet 3.0: an enhanced lexical resource for sentiment analysis and opinion mining.” Lrec. Vol. 10. No. 2010. 2010.

[2] George A. Miller (1995). WordNet: A Lexical Database for English. Communications of the ACM Vol. 38, No. 11: 39-41.

[3] Mingzhi, Cheng, et al. ”A random walk method for sentiment classification.” Future Information Technology and Management Engineering, 2009. FITME’09. Second International Conference on. IEEE, 2009.

[4] Turney, Peter D., and Michael L. Littman. ”Unsupervised learning of semantic orientation from a hundred-billion-word corpus.” arXiv preprint cs/0212012 (2002).

[5] Hassan, Ahmed, et al. ”A random walk-based model for identifying semantic orientation.” Computational Linguistics 40.3 (2014): 539-562.

[6] Xu, Yunpeng, Xing Yi, and Changshui Zhang. ”A random walks method for text classification.” Proceedings of the 2006 SIAM International Conference on Data Mining. Society for Industrial and Applied Mathematics, 2006.

[7] Islam, Md Rafiqul, and Md Rakibul Islam. ”An improved keyword extraction method using graph based random walk model.” Computer and Information Technology, 2008. ICCIT 2008. 11th International Conference on. IEEE, 2008.

[8] Mingzhi, Cheng, et al. ”A random walk method for sentiment classification.” Future Information Technology and Management Engineering, 2009. FITME’09. Second International Conference on. IEEE, 2009.

[9] Yazdani, Majid, and Andrei Popescu-Belis. ”A random walk framework to compute textual semantic similarity: a unified model for three benchmark tasks.” Semantic Computing (ICSC), 2010 IEEE Fourth International Conference on. IEEE, 2010.

[10] Jiang, Shan, and ChengXiang Zhai. ”Random walks on adjacency graphs for mining lexical relations from big text data.” Big Data (Big Data), 2014 IEEE International Conference on. IEEE, 2014.

[11] Xu, Guandong, Bin Fu, and Yanhui Gu. ”Point-of-interest recommendations via a supervised random walk algorithm.” IEEE Intelligent Systems 1 (2016): 15-23.

[12] Hassan, Ahmed, and Dragomir Radev. ”Identifying text polarity using random walks.” Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics. Association for Computational Linguistics, 2010.

[13] Montejo-Raez, Arturo, et al. ”Random walk weighting over sentiwordnet for sentiment polarity detection on twitter.” Proceedings of the 3rd Workshop in Computational Approaches to Subjectivity and Sentiment Analysis. Association for Computational Linguistics, 2012.

APPENDIX

A. LINK TO CODE

https://tinyurl.com/y8umpu44

B. CONTRIBUTION

- **Ekta Rita** - Data Extraction and Selection code, Similarity Matching Implementation, NetworkX Graph Design, PageRank Implementation, Visualization of Result
- **Shreya Venkatesh** - Data Extraction for Random Walk, Literature Survey, Graph Representation Design and Implementation, Code for Results aggregation
- **Srisha Raviillu** - Code for : Data Extraction and Selection, Attribute formatting for Similarity Comparison, Evaluation of Results, Visualization