Business: Do you wanna sell more? Discovering Topics, Sentiments and Prediction of Ratings

Mitodru Niyogi
Govt. College of Engineering & Ceramic Technology
Kolkata, West Bengal
mitodru.niyogi@gmail.com

Asim Kumar Pal
Management Information Systems
Indian Institute of Management Calcutta
Kolkata, West Bengal
asim@iimcal.ac.in

ABSTRACT

In the era of Social Computing, the role of customer reviews and ratings can be instrumental in predicting the success and sustainability of businesses as customers and even competitors use them to judge the quality of a business. Yelp is one of the most popular websites for users to write such reviews. This rating can be subjective and biased toward user's personality. Business preferences of a user can be decrypted based on his/ her past reviews. In this paper, we deal with (i) uncovering latent topics in Yelp data based on positive and negative reviews using topic modeling to learn which topics are the most frequent among customer reviews, (ii) sentiment analysis of users' reviews to learn how these topics associate to a positive or negative rating which will help businesses improve their offers and services, and (iii) predicting unbiased ratings from user-generated review text alone, using Linear Regression model. We also perform data analysis to get some deeper insights into customer reviews.

KEYWORDS

Topic modeling, sentiment analysis, predictive analysis, data visualization, text mining, Yelp reviews, machine learning

ACM Reference format:
Mitodru Niyogi and Asim Kumar Pal. 2017. Business: Do you wanna sell more? Discovering Topics, Sentiments and Prediction of Ratings. In *Proceedings of International Conference on Computer and Communication Technology, Allahabad, India, November 24–26, 2017 (ICCCT-2017),* 6 pages.
https://doi.org/10.1145/3154979.3154987

1 INTRODUCTION

Topic models [1] are a way to discover underlying themes in an otherwise unstructured collection of documents. In this study, we specifically used the Latent Dirichlet Allocation (LDA) [2] based topic model on a dataset of Yelp reviews to cluster restaurants based on positive and negative reviews. For text categorization [3], we have clustered the review stories into several k topics: unsupervised learning with automatic topic labeling i.e., topic modeling. It gives insights into uncovering hidden patterns or unknown co-relations from text reviews. LDA treats data as observations that arise from a generative probabilistic process including hidden variables for documents which reflect the thematic structure of the collection. Inferring them using posterior inference results into topic generation that describes its corpus.

Talking about sentiment analysis which assigns a sentiment association score to each story depending on the expressive tone of the review. Sometimes, a user review is biased towards personality, preference and idiosyncratic behavior of a particular reviewer. This might affect overall ratings of a business. In this paper we also try to decode users' preferences on users' past reviews using predictive modeling technique.

User's restaurant preferences can be decrypted based on their past restaurant reviews. The experiment involves evaluating different feature extraction methods: Term Frequency (TF) and Term Frequency - Inverse Document Frequency (TF-IDF) [4] of review text. We also train extracted features using machine learning techniques such as Linear Regression for rating prediction and then record the Root Mean Square Error(RMSE) [5] . We use the Root Mean Square Error to quantify our error, instead of using accuracy.

The road-map of the paper is as follows. Data preparation and data analysis on Yelp data are described in section 2. Section 3 is on background. Section 4 reveals our proposed system architecture. Section 5 deals with the experiment setup. Section 6 gives the results. Section 7 draws conclusions from discussions and points to future work.

2 DATA SET

In this paper, we experiment with the dataset provided by Yelp [6] for exploratory, topics clustering, and sentiment analysis as well as for training and testing the unbiased rating prediction model. The dataset includes records from US, UK, Canada and Germany. It contains information on businesses, business attributes, check-in sets, tips and text reviews. The dataset consists of five JSON files, namely: business, review, user, check-in, and tip JSON objects. Each file consists of one json-object-per-line. Thus, a business is represented in the 'business.json' file as a json object which states the business ID, its name, location, stars, review count, opening hours, etc. Similarly, a text review is a JSON object in the 'review.json' file, which specifies the business ID, user ID, stars (integer values between and including 1 and 5), review text, date and votes. In our study, we deal with business.json, review.json and check-in.json files, and we do not use the rest of the data. The businesses informations in the Yelp dataset belong to diverse categories, such as restaurants, shopping, hotels and travel, etc. The text reviews for different variants of business categories may be very different. We also perform some data analysis on the overall dataset before preprocessing steps for our model.

ICCCT-2017, November 24–26, 2017, Allahabad, India

ACM ISBN 978-1-4503-5324-3/17/11…$15.00
https://doi.org/10.1145/3154979.3154987

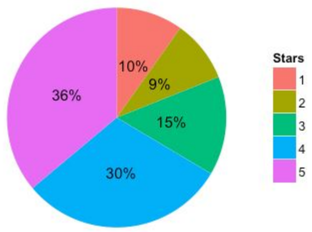

Figure 1: Restaurant review ratings distribution.

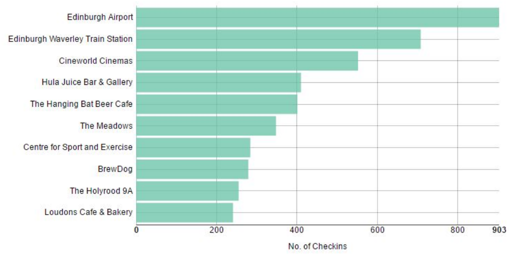

Figure 2: Top 10 Business in UK by Checkins

2.1 Data Analysis

Fig. 1 represents review ratings distribution of the entire dataset. We see more than 60% reviews have higher ratings of 4 stars or more and the rest contributes to poor ratings reviews. Fig. 2 ranks top 10 businesses in UK by number of check-ins.

2.2 Data Preprocessing

The data from the NoSQL [7] database was imported into R console using the tm package [8] in CRAN library to construct the document term matrix for use in developing the topic model. We write some basic R scripts to separate the restaurants from the business.json file, and to separate the restaurant reviews from the review.json file. We then preprocess the text reviews as follows. Stop words were removed and stemming was done to extract meaningful content from each of them. To do this, we use standard R CRAN 'tm' library [8] to remove capitalizations, stop words [9] and punctuations. The following section describes more in details about our feature extraction approaches.

2.3 Feature Extraction

We use two methods to abstract useful features from the review text corpus and to build a feature vector for each review. Each of these methods relies on semantic analysis of the text.

2.3.1 Unigrams. In the uni-grams model (also called the "bag of words" model) [10], each unique word in the review text is considered as a feature. Thus, building a feature vector for a review text is upfront. Initially, we create a dictionary of the entire vocabulary of words present in the review corpus. Then a word-review matrix is created, where entry (i,j) is the frequency of occurrence of word i in the j^{th} review text. Lastly, the TF-IDF [4] weighting technique is applied to this matrix to derive the final feature matrix. As we know, this weighting technique assigns less weight to frequently occurring words across reviews (e.g. "meal").

2.3.2 Bigrams. The inability of the unigram model to capture relationships between two words in pair as it treats each word individually. Thus, we add bigrams [10] to the unigram model to capture the effects of the pair words such as 'tasty pizza' and 'not yummy'. Now, the dictionary is expanded with more rows to accommodate the additional 2-tuples of words (i.e. all pairs of consecutive words) occurring in the corpus of reviews text. The matrix is computed and TF-IDF weighting is applied to this matrix as before.

3 BACKGROUND

3.1 Introduction to LDA

Previously, documents were treated as "a-bag-of-words" [11] approach as in many models which dealt with text documents. Topic modeling adopts that a document is "a-bag-of-topics" instead of "a-bag-of-words" representation, and its sole purpose is to cluster each term in each document into a relevant topic. A variations of different probabilistic topic models [12] have been proposed and LDA [2] is considered to be a well known method. Alike other methods, the input to LDA is a term × document matrix, and the output of LDA is composed of two distributions, namely document-topic distribution θ and topic-word distribution ϕ. EM [13] and Gibbs Sampling [14] algorithms were proposed to derive the distributions of θ and ϕ . In this paper, we use the Gibbs Sampling based LDA. In this approach, one of the most significant step is updating each topic assignments individually for each term in every documents according to the probabilities calculated using Equation 1.

$$P(z_i = k \mid w_i = v, z_{-i}, w_{-i}) \propto \frac{C^{WT}_{vk} + \beta}{\Sigma_{v'} C^{WT}_{vk} + V\beta} \cdot \frac{C^{DT}_{dk} + \alpha}{\Sigma_{k'} C^{DT}_{dk} + K\alpha} \tag{1}$$

where z_i=k represents that the i^{th} term in a document is assigned to topic k, w_i =v is the mapping of the observed term w_i to the v_{th} term in the corpus's vocabulary, and z_{-i} signifies all the assignments of topic except the i_{th} term. C^{WT}_{vk} is the frequency of occurrence of term v assigned to a particular topic k, and C^{DT}_{dk} is the number of times that the document d contains the topic k. Moreover, K is the user input denoting the number of topics, V represents the vocabulary's size, hyper-parameters for the document-topic distribution and topic-word distribution are denoted by α and β respectively. By default, α and β are set to 50/K and 0.01.

We perform N iterations of Gibbs sampling for every terms in the corpus and after this, we estimate the document-topic θ and topic-word ϕ distributions respectively using Equations 2 and 3.

$$\theta_{dk} = \frac{C^{DT}_{dk} + \alpha}{\Sigma_{k'} C^{DT}_{dk} + K\alpha} \tag{2}$$

$$\phi_{vk} = \frac{C^{WT}_{vk} + \beta}{\Sigma_{v'} C^{WT}_{vk} + V\beta} \tag{3}$$

3.2 NMI

In the experiment we used NMI (Normalized Mutual Information) [15] to evaluate overall documents cluster quality. The following formula is used to calculate NMI:

$$NMI(X,Y) = \frac{2I(X,Y)}{H(X) + H(Y)} \tag{4}$$

where I(X;Y) is mutual information between X and Y, where X = X1, X2, ...Xn and Y = Y1, Y2,...Yn. Xi is the set of text reviews in LDA's topic i while Yj is the set of text reviews with the label j. In our experiments, a text review with the label j means that the text review has the highest probability of belonging to topic j; n is the number of topics. I(X;Y) is

$$I(X,Y) = \sum_{y \in Y} \sum_{x \in X} p(x,y) \log(\frac{p(x,y)}{p(x)p(y)}) \tag{5}$$

In the formula, $p(x_i)$ means probability of being classified to topic i, $p(y_j)$ means probability of labeled to topic j while $p(x_i,y_j)$ means probability of being classified to cluster i but actually labeled to cluster j. H(X) is entropy of X as calculated by the following formula:

$$H(X) = -\sum_{i=1}^{n} p(x_i) \log p(x_i) \tag{6}$$

The clustering result is totally different from the label if the value of NMI is 0 and is identical if value of NMI is 1.

3.3 Sentiment Analysis

Sentiment analysis [16], also called opinion mining, is the field of study that studies people's opinions, sentiments, evaluations, appraisals, attitudes, and emotions towards entities such as products, services, organizations, individuals, issues, events, topics, and their attributes.

3.4 Prediction of Rating

In this study, the job is to predict rating from review text alone. Having the bag of words, we treat the problem of prediction of a business rating star as a regression problem. As our objective is to predict the business' star using regression model, we use the Root Mean Square Error [5] to quantify our error, instead of using accuracy. The equation is shown below:

$$RMSE = \sqrt{\frac{1}{n} * \sum_{j=1}^{n} (y_j - \overline{y})} \tag{7}$$

4 PROPOSED SYSTEM ARCHITECTURE

We propose a system that consists of three main components including data collection, data analysis and data visualization. The data collection module is developed to crawl the "Restaurant" category data using crawler and to store the data into MongoDb [7], a NoSQL database for scalability and scheme less data storage purpose. After data preprocessing steps such as tokenization, stemming and stopwords removal, the system mainly performs different types of analyses to answer the following questions:

- What are the latent topics written by customers in text reviews to help business understand customer's feedback?
- Why is a particular restaurant business hit? What are the primary traits of a restaurant business to attract customers? Why do customers complaint? Why were customers not satisfied by the service offered by businesses? What are people's opinion on the specific restaurant business that can be helpful in understanding customers' positive, negative or neutral reactions?
- Are these reviews biased? How to predict unbiased ratings from users' text reviews?
- What are the correlated topics that occur together online to help us understand the trends and patterns of user's interest?

4.1 Approach

The term-document matrix is created which is fed to LDA based model for discovering latent topics and the documents are analyzed by the sentiment analyzer. Then, sentiment analyzer will assign each text review with associated AFINN score on a scale of -5 to +5 to classify reviews as positive, negative or neutral. Fig. 3 presents the architecture of our proposed system. Then, using the "bag of words" representation of text reviews, a predictive model is being trained using Linear Regression model to predict the unbiased ratings from users' text alone. RMSE is used to quantify the error between predicted value and original, instead of using accuracy.

For sentiment analysis, we use AFINN-111 [17] in our experiment. AFINN is a list of English words rated for valence with an integer between minus five (negative) and plus five (positive). The words in the sentiment lexicons of AFINN have been manually labeled.

For the predictive task, we use Linear Regression to train our model. We used the term frequency of top 2000 unigrams across all reviews to train the model. The same was trained using the top 2500 bigram words. We further enhanced the linear regression model by using term frequency - inverse document frequency of top 2000 unigrams in the review text. Fig. 3 depicts our proposed model architecture.

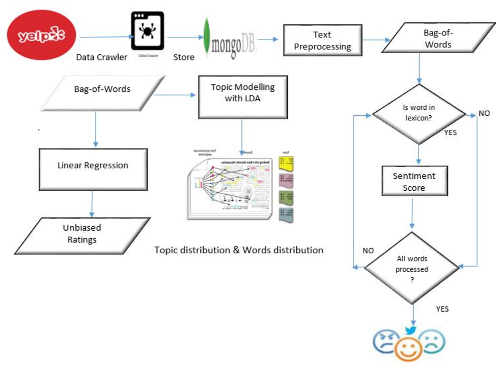

Figure 3: Proposed Model Architecture

5 EXPERIMENTAL SETUP

For the Yelp data, we started with default parameters α = 0.1; β = 0.01 and input parameter topic number N =5, 10, 15, 20 which means 5, 10, 15, 20 desired topics. By comparing the LDA result given in Table 1, we choose topic number N = 20 as a basic group for further

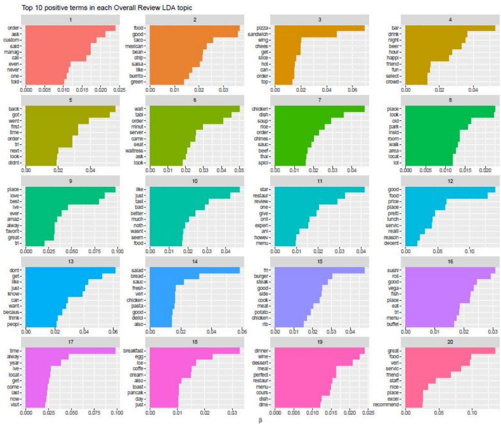

Figure 4: Topic 10 terms in overall topic analysis

comparison since when N = 20, most topics have enough words to reveal information about the topic while without too much words to make the topics messy. In the next step of our experiment, we set N = 20 and tuning parameter α and β by setting α = 0.1, 0.05, 0.2 while β = 0.01, 0.015, 0.007 to see if the results show any difference.

Table 1: NMI of models

LDA Models	NMI Results
YelpRw(5,0.1,0.01)	0.548
YelpRw(10,0.1,0.01)	0.588
YelpRw(15,0.1,0.01)	0.590
YelpRw(20,0.1,0.01)	0.463

The overall topic model lists a number of topics about customer experience. But good and bad experiences are mixed because the corpus consist of both positive and negative reviews. We explore the topics related to positive and negative ratings independently.

We have built-in two topic models (one for positive reviews and another for negative reviews) with 20 topics each and following the same approach that we used to compute the overall topic model. We didn't take account of reviews with three stars because that rating is neither positive nor negative. To avoid an uneven number of reviews in each category, we use a random sample for the positive grouping with size equal to the number of reviews in the negative category.

Broadly speaking, we can see some of the overall topics also appear in these two new models. These new models also show a finer grain decomposition of the customer experience.

6 RESULTS

6.1 Topic Analysis

Fig. 4 shows the distribution of top 10 terms in overall topic distribution. Table 2 lists the words distribution in the overall topic analysis.

Positive topics analysis: We see that many topics from overall topics overlap in positive ratings criteria. Top food quality, ambience, service quality, prime location restaurants, cuisine food flavor,

Table 2: Overall topic analysis

Topic No:	Description	Respective Words
Topic 1	poor service	order, ask, call, never, manage
Topic 2	Mexican cuisine	food, good, taco, burito
Topic 3	Italian cuisine	pizza, sandwich, wings
Topic 4	weekend bar	bar, drink, night, beer, happy
Topic 5	waiter service	wait, table, order, came, late
Topic 6	lunch menu	chicken, dish, rice, thai, noodles
Topic 7	ambience	place, look, old, shabby
Topic 8	great place	place, amaze, love, great
Topic 9	poor food quality	poor, taste, bad, food
Topic 10	1 star rating	star, one, menu, reviews
Topic 11	dinning traits	food, place, service, decent
Topic 12	Friday's top menu	burger, meat, streak, meat, bbq
Topic 13	Japanese cuisine	sushi, shrimp, seafood, roll
Topic 14	breakfast menu	egg, coffee, toast, pancake
Topic 15	high demand	time, location, positive
Topic 16	dinner	dinner, wine, desert, delicious

weekend party, bars and hot favorite prime dishes of variety of restaurants are the most conversed topics about businesses contributing to positive ratings . Words like "awesome", "delicious", "recommend", "nice staff", "enjoyable", "family dinner", "visit again", "cooked perfectly", "location great" have been dominating across positive reviews with ratings of more than 3 stars; "chicken, steak, taco, sushi, pizza, burger, burrito, roll, sandwich, bacon, soup, salad, beer" are the most popular dishes as breakfast, lunch, dinner platter across text reviews.

Negative topics analysis: Poor hygiene, poor hospitality poor food quality, high price, poor customer management, waitress service, odd locations, etc., are some topic conversations that people wrote in their reviews. Words like "disgust", "rude", "terrible", "cold", "never visit", "wait", "serve late", "food service bad", "one star", "long hour", "horrible", "uncooked", "denied", "racist", "worthless penny" are observed in negative reviews topics of our model.

Fig. 5 displays some topics word cloud of positive ratings. Fig. 6 displays some topics word cloud of negative ratings. Fig. 7 displays some topics word cloud of overall corpus topic analysis.

However we also see that some people were quite biased on their reviews, they have liked the food but didn't like the place so they rated poor; at sometimes they have enjoyed breakfast menu but the tea was served cold or they were kept waiting for long, resulted into poor ratings of restaurants despite their quality menu. We also see that people devalued the quality of food served for higher price; high price high food mantra has not been appreciated rather okayish food quality at less price has been rated with higher stars than the former. We do see a mixed set of reactions on same parameters. At a particular place someone rated high for ambience, on contrary other rated low for being expensive. This became our motivation to predict a business' rating based on its reviews text alone to reduce the bias of the reviewers.

Figure 5: Topics on Positive Ratings

Figure 6: Topics on Negative Ratings

Figure 7: Overall topics

6.2 Sentiments

We can see in Fig. 8 that word such as "bad" with AFINN score of -3 has appeared the most number of times in the reviews, and some words, like "damn", are often positive (e.g., "the grill chicken was damn good!"). Some of the words that AFINN most underestimated included "die" ("the pork chops are to die for!"), "joke" ("the service is a complete joke!").

Fig. 8 shows a plot of ratings against sentiments. This plot gives insights about the AFINN sentiment score of words used in text reviews against particular ratings. Table 3 lists positive and negative respective words.

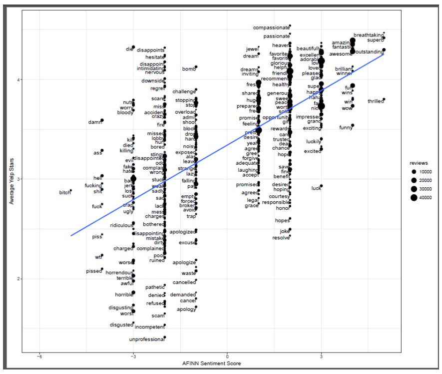

Figure 8: Ratings v/s Sentiments

Table 3: Sentiments Distribution

Sentiments	Representative Words
Positive	Ratings 4 and above : compassionate, superb, heavenly, fantastic, bliss, perfect, awesome, lovely, love, favorite, yummy, joy, friendly, recommend Ratings 3: attraction, hope, desire, luckily, careful, sincerely, helping
Negative	Rating 2 and less: disgust, hate, insult, refused, cancel, unprofessional, worst, horrible, awful, scam, pathetic, ridiculous, ugly, depressing, irritating, nasty, dirty, gray, lame, appalled, refund

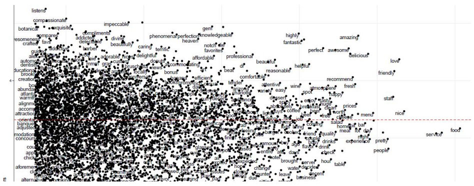

Figure 9: Words for positive sentiments

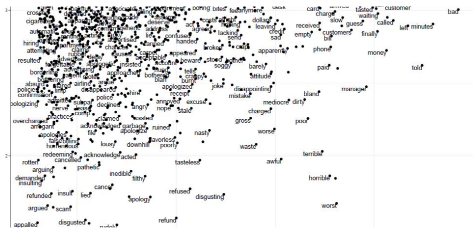

Figure 10: Words in Negative Reviews sentiment

Missclassification: In AFINN sentiment analysis we have seen that some negative words like "disappoints", "bomb", "downside", "hesitant", "bloody" have been classified to positive sentiments with high AFINN score. However, the word "bomb" can sometimes be used as positive connotation in natural language. Fig. 11 shows the misclassification.

6.3 Prediction

Evaluation Metrics: Our aim in linear regression(LR) is to reduce the Root Mean Squared Error (RMSE) of the predicted values of the business star ratings when in comparison to the baseline model.

Baseline: The baseline model will always predict the average star rating of all the reviews in the dataset.

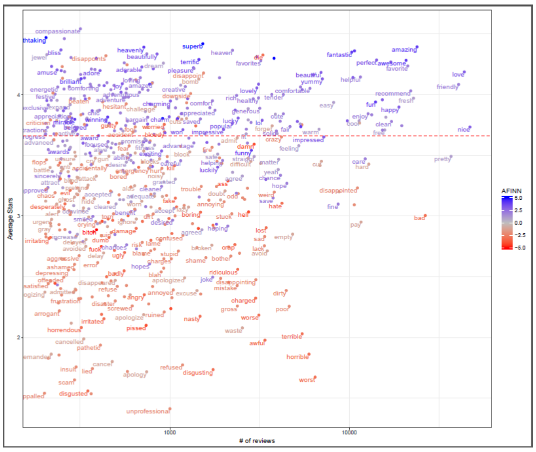

Figure 11: Misclassification of Sentiments v/s Ratings

The RMSE of the LR model was less than the unigram model which used only term frequency as a weightage. But the RMSE again increases when use a bigram model with term frequency of top 2500 bigrams. Hence, it seems from this analysis, we are able to claim that the linear regression model works best when we use term frequency - inverse document frequency of top 2000 unigrams.

Table 4: MSE Error measures for predictive model

Model	Feature	RMSE
Baseline		1.34047969
Linear Regression	Unigram + TF	0.909073979
Linear Regression	Bigram + TF	1.046983524
Linear Regression	Unigram + TF-IDF	0.815036502

7 DISCUSSIONS AND CONCLUSION

We visualized the latent topics in the form of word clouds as well as derived adequate insights from graphical plots. In the experiment with Yelp reviews, we have explored some latent topics in overall corpus of reviews for restaurants by fitting a topic model to the corpus using LDA with Gibbs sampling. The topics found related to themes of different cuisines and customer experiences. We further explore the customer experience topics by splitting the corpus in two corpora, one for positive experiences i.e., ratings of business greater than three on a scale of five; another for negative experiences (less than three stars) by fitting a topic model to each corpus. There has been an overlap of themes in the new models based on ratings with overall topic model. The new models illustrate a finer grain disintegration of the customer experiences based on positive and negative ratings of user reviews in USA and UK. The LDA Gibbs has been able to distinguish latent topics on Yelp data in JSON format quite well.

Meanwhile, in the sentiment analysis of Yelp Reviews data in USA, we have been able to extract sentiments using AFFIN score on a scale of 10 from -5 to +5 for important words used by users to make a positive or negative impact on reviews. This gives an insight to business companies about the quality of service that they are providing to their customers without reading the millions of reviews sequentially and which are the words emphasized by users in written reviews impacting the ratings of restaurants or other business in Yelp. Our exploration of data through sentiment analysis and topic modeling which clustered the reviews based on positive and negative reviews empowers businesses to gain insights for marketing, advertising, quality, supply chain management, customer retention, etc.

Lastly, as user-generated reviews suffer from subjectivity and being biased toward users' personality, we consider Linear Regression models in order to predict the user rating from review text corpus. It is revealed that Linear Regression model with TF-IDF and unigram features is found to give the least RMSE.

In future work, we can extend our model to integrate rating of restaurants predicted as features to perform user restaurant recommendations by clustering. We can also extend our model to a emotion grained constrained topic model which will cluster topics with respect to each basic human emotions. We can also use other machine learning techniques such as Decision trees, Gradient Boosting as a baseline model for our predictive task in future.

REFERENCES

[1] Hanna M. Wallach. 2006. Topic modeling: beyond bag-of-words. In Proceedings of the 23rd international conference on Machine learning (ICML '06). ACM, New York, NY, USA, 977-984. DOI: https://doi.org/10.1145/1143844.1143967

[2] Blei, David M., Andrew Y. Ng, and Michael I. Jordan. 2003. Latent Dirichlet Allocation. *Journal of Machine Learning Research* 3:993-1022.

[3] Srivastava, A., Sahami, M. (eds.) Text mining: Classification, Clustering and Applications, pp. 155-184. CRC Press, Boca Raton, FL.

[4] Yates, B. R., Neto, R. B. (1999) Modern Information Retrieval, ACM Press, New York.

[5] Fan, Mingming, and Maryam Khademi. "Predicting a business star in Yelp from its reviews text alone." arXiv preprint arXiv:1401.0864, 2014.

[6] "Yelp 2014 Dataset Challenge." [Online]. Available: http://www.yelp.com/dataset_challenge/. Accessed: July 30, 2014.

[7] MongoDB-MongoDB, Inc. https://www.mongodb.com/.

[8] Text Mining Package in R. https://cran.r-project.org/web/packages/tm/tm.pdf.

[9] Porter, M. F. (1980). An algorithm for suffix stripping. In *Program*, volume 14, pages 130-137.

[10] Christopher D. Manning, Prabhakar Raghavan, Hinrich Schütze: An Introduction to Information Retrieval, pages 237–240. Cambridge University Press, 2009

[11] Zhai Z., Liu B., Xu H., Jia P. (2011) Constrained LDA for Grouping Product Features in Opinion Mining. In: Huang J.Z., Cao L., Srivastava J. (eds) Advances in Knowledge Discovery and Data Mining. PAKDD 2011. Lecture Notes in Computer Science, vol 6634. Springer, Berlin, Heidelberg

[12] Blei, David M. 2012. Probabilistic topic models. *Communications of the ACM* 55 (4):77-84.

[13] T. L. Griffiths and M. Steyvers. Finding scientific topics. *Proceedings of the National Academy of Sciences*, 101(suppl 1):5228-5235, 2004.

[14] T. Griffiths, "Gibbs sampling in the generative model of latent dirichlet allocation", 2002.

[15] Mehrotra, R., S. Sanner, W. Buntine, L. Xie, Improving LDA Topic Models for Microblogs via Tweet Pooling and Automatic Labeling, *Proceedings of the 36th international ACM SIGIR conference on Research and development in information retrieval.* 2013.

[16] Bo Pang and Lillian Lee. 2008. Opinion Mining and Sentiment Analysis. Found. Trends Inf. Retr. 2, 1-2 (January 2008), 1-135. DOI=http://dx.doi.org/10.1561/1500000011

[17] Lars Kai Hansen, Adam Arvidsson, Finn Årup Nielsen, Elanor Colleoni, Michael Etter, "Good Friends, Bad News - Affect and Virality in Twitter", The 2011 International Workshop on Social Computing, Network, and Services (SocialComNet 2011).