Mining Opinion Features in Restaurant Reviews

Wen Cui, Wenjia Ma, Xilian Li
University of California, Santa Cruz
{wcui7,ada.ma,xli237}@ucsc.edu

ABSTRACT

More and more people would like to choose preferred restaurants based on online reviews, and share opinions through online applications, with the development of online technologies. however, most of previous research focused on how to generate overall ratings or opinions from online reviews, like positive or negative. This paper proposed a new framework to generate opinions of specific features extracted from restaurant reviews to help people find their preferred restaurants quickly, to improve people's experience of online service, by combining lexical analysis, feature extraction, feature classification, and feature-based sentiment analysis.

Keywords

Natural language processing; Feature-based opinions; Feature extraction; AFFIN Sentiment Score; Tf-idf; Word Embedding; Stanford PCFG parser model

1. INTRODUCTION

With the emergence of online user-generated content platform, like Yelp.com, more and more people would like to make decisions based on online information, like how to choose a preferred restaurant according to the restaurantâĂŹs online reviews.

Compared to the detailed official descriptions of each restaurant, customers preferred to believe in actual customers' reviews. However, popular restaurants generally have too many reviews to read one by one. While customers generally share reviews and ratings according to their own preferences and aspects, in order to find the most suitable restaurant, people have to extract useful information from as many reviews as possible, because people always care about and have specific preferences for the restaurant. Extracting all useful information manually from online restaurant reviews can be impossible to complete for many popular restaurants, which have hundreds or thousands of reviews. If online service platforms can provide the service of generating feature-based opinions from each restaurant's reviews, people can quickly make a decision whether this restaurant is suitable, as shown in the effect picture in Figure 1.

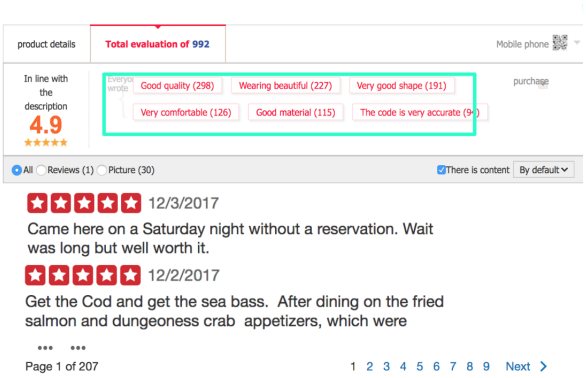

Figure 1: The effect picture

Furthermore, feature-based opinions can also help restaurants figure out how to improve themselves with respect to specific aspects which they actually need to improve, according to customers' feedback. Besides, feature opinions extracted from restaurant reviews can also be extended and applied into restaurant recommendations, reviews assessment, menu generation and feature dishes.

Thus, this paper aims to propose a new framework to generate feature-based opinions from online restaurant reviews, by combining lexical analysis, feature extraction, feature classification, and feature-based sentiment analysis. We performed lexical analysis using Stanford PCFG parser[6], and then extracted feature patterns from these reviews. Before further analysis, we consider feature pruning to filter 46 percentage of the data and keep subjective feature patterns[8]. After that, we classify these feature patterns into four categories, DECOR, FOOD, SERVICE and OTHER, using Decision Tree classifier, with n-fold cross validation.[9] Eventually, we can use AFINN sentiment score[12] to summarize each specific feature's opinions.[1]

2. RELATED WORK

Research in text data mining has produced a variety of tools to deal with documents, which can also be used to mine

[1]Check our github page https://github.com/wenzi3241/TIM245-Course-Project for source code.

© 2017 ACM. ISBN 978-1-4503-2138-9.
DOI: 10.1145/1235

opinion features from restaurant reviews. However, most of previous research on restaurant reviews focused on the overall sentiment analysis of each restaurant review[1], like how to determine whether the review is positive or negative[2]. However, overall sentiment analysis cannot indicate detailed feature opinions which customers care more about, according to their own preferences. Recently, some researchers studied how to extract feature from online reviews[3], which provides useful tools for feature-based opinion summarization. The Standord parser [6] also provide useful tools for text analysis. More and more people started to focus on how to summarize opinion features from restaurant reviews. This paper proposed a new framework to generate feature-based opinions from online restaurant reviews, by combining lexical analysis, feature extraction, feature classification, and feature-based sentiment analysis.

3. DATASET

The yelp challenge dataset has more than two million reviews, and 50 percentage of the reviews have more than one hundred words, as shown in Figure 2. The distribution of the length of each review can also indicates necessity of feature-based summarization.

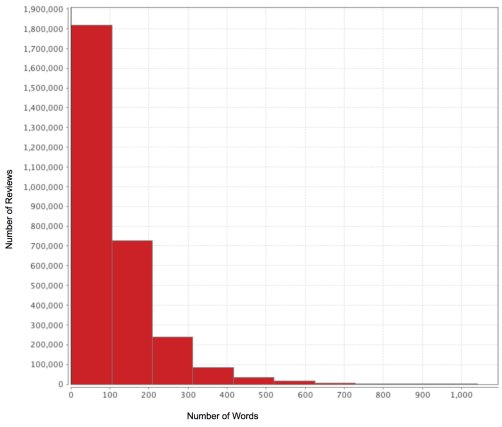

Figure 2: Length Distribution of each review

We obtained 1,048,576 reviews from 21,835 restaurants from the Yelp Challenge dataset. After analysis and visualization, we found almost 90 percent of the reviews were concentrated on 10 percent of restaurants in Vegas, as shown in Figure 3. That is to say, the average number of reviews of each top 10 percent restaurant can be up to more than 500. Generally speaking, it's just the top 10 percent of the restaurants which make people hard to determine which one is better than the others. Therefore, further research and analysis need to focus on restaurants that have more than five hundred reviews.

4. SYSTEM OVERVIEW

4.1 Preprocessing

4.1.1 *Data cleaning*

The yelp dataset is quite complete whereas no missing data presented. However we mainly focus on mining the reviews data which are closer to spoken language even though

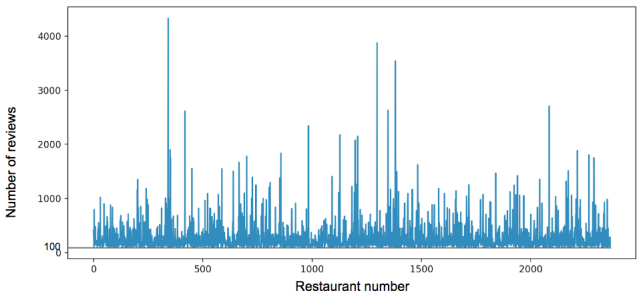

Figure 3: Distribution: number of reviews from different restaurants

they present in the form of written language. So in this step, we process the data by removing special characters like :), :), p, etc which may have emoji meaning but potentially jeopardize the parse tree(discuss in Section 4.1.3) of the sentence.

4.1.2 *Data reduction*

Numerosity reduction techniques replace the original data volume by alternative, smaller forms of data representation [4]. In order to discover interesting patterns, we form a subset by keeping restaurants with more than 500 reviews. We can visualize one of the restaurant(named Table 17) reviews in Figure 4 generated by word cloud [5]. The most frequent words people used are *'Table', 'good', 'great', 'food'* etc. We can see the potential information we can mine from such review data.

Figure 4: Word cloud of restaurant 'Table 17' with 500 reviews

4.1.3 *Natural Language Process - Parsing*

A natural language parser is a model that works out the grammatical structure of sentences, for instance, which groups of words go together (as "phrases"). Probabilistic parsers use knowledge of language gained from hand-parsed sentences to try to produce the most likely analysis of new sentences. These statistical parsers still make some mistakes, but commonly work rather well. Their development was one of the biggest breakthroughs in natural language processing in the 1990s. [6]

We run the Standford a lexicalized PCFG parser(which yields a good performance statistical parsing system) on the reviews. An example of parsing the sentence *'The steak frites are the best in the city.'* is shown in Figure 5. As de-

sired, *'the steak frites', 'the best', 'the city'* are grouped together as NP (stands for noun phrase) and *'best'* is tagged as JJS(Penn Treebank for adjective [7]). These are the essential to our feature extraction discussed in Section 4.2. Also we can see the punctuation acts as a node in the parse tree so that the data cleaning process discussed earlier is important in the sense of producing more accurate trees.

```
(ROOT
 (S
  (NP (DT The) (NN steak) (NNS frites))
  (VP (VBP are)
   (NP
    (NP (DT the) (JJS best))
    (PP (IN in)
     (NP (DT the) (NN city)))))
  (. .)))
```

Figure 5: An example of parse tree

4.2 Feature Extraction

We read through some of the reviews and find out when people give opinions, they tend to use sentence structure like *'The environment is nice.', 'The food is delicious.'*. The attributes(or aspects) of the restaurant are expressed mostly as the single noun or noun phrases in the sentence. And the opinion words mostly are adjective. Although there exist other forms of expressing opinion as we observed such as *'beyond expectation', 'never come back again'*. Better rules are needed in order to capture these language patterns.
From the above observation, we extract noun phrases and adjective (which is closed to the head of noun phrases) as (attribute, opinion) pair. And also store the negation word(if present) as it changes the polarity of opinions.

4.3 Feature pruning

We mainly consider two ways of pruning. One is to set a simple frequent threshold. As we get our candidate (attribute, opinion) pairs by keeping only attributes appearing more than certain threshold(5 in our experiment). Another way is classifying the review sentences into subjective and objective. Considering subjective review mainly express one's opinion. In this step, we aiming to get rid of non-relevant attributes such as location, story-telling, etc. For example *'This place is located near Beverly Street.', 'I went there with my husband for our anniversary.'* would be classified as objective and no desired or interesting attributes can be found. Whereas *'I left Table 17 feeling very ambivalent.'* would be classified as subjective with interesting opinion pattern.
We use an off-the-shelf tool to do the subjective/objective classification(can be found in here [8]) since our goal of this project is not such classification. And this process significantly filters 46% of data.

4.4 Classification

This process serves as producing a good summary of reviews. In order to do that, we classify the attributes extracted from previous sections into four categories. In this section, we will discuss how to get the training data and the feature selection for the classification.

4.4.1 Training set

We manually annotate 780 candidate attributes extracted from previous steps, and put them into four categories DECOR, FOOD, SERVICE and OTHER. The annotations are entirely done by Xilian Li, one of our group members, in order to generate a better understanding and consistent labeling. The distribution of labels is shown in Figure 6.

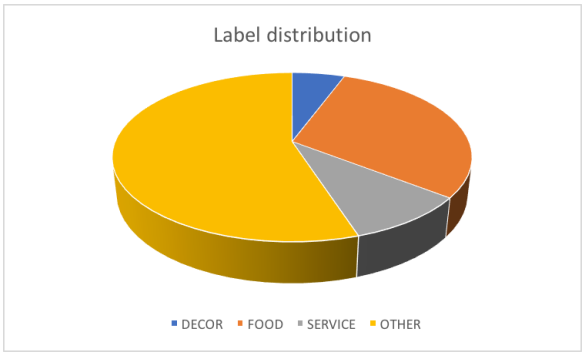

Figure 6: Distribution of four categories

4.4.2 Decision Tree Classification

We implement decision tree classification by using the library from scikit learn [9]. There are several reasons we choose this model,

- Simple to understand and to interpret. Trees can be visualized.
- Requires little data preparation. Other techniques often require data normalization, dummy variables need to be created and blank values to be removed.
- Low predicting time. The cost of using the tree is logarithmic in the number of data points used to train the tree. Which could benefit if we want to transfer the system into a real time system.
- Ability to handle multi-label problem.
- Uses a white box model. If a given situation is observable in a model, the explanation for the condition is easily explained by boolean logic. By contrast, in a black box model (e.g., in an artificial neural network), results may be more difficult to interpret.
- Possible to validate a model using statistical tests. That makes it possible to account for the reliability of the model. Which will be discussed in Section 4.5.

Figure 7 is a subtree of our model with word embedding. The complete tree can be viewed through our Github page.

4.5 Evaluation

In this section, we compare the decision tree model performance using 10-fold cross validation with different kinds of feature selection techniques.

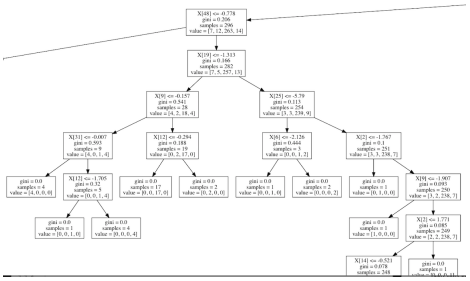

Figure 7: Decision Sub-tree with word embedding

4.5.1 *Tf-idf*

Firstly, we apply the most widely used term-weighting schemes: tf-idf. Which shorts for term frequency-inverse document frequency, is a numerical statistic that is intended to reflect how important a word is to a document. The tf-idf value increases proportionally to the number of times a word appears in the document, but is often offset by the frequency of the word in the corpus, which helps to adjust for the fact that some words appear more frequently in general (for example stopwords).

4.5.2 *Word Embedding*

Secondly, we implement word embedding as a representation of our data. And word embedding is a language modeling and feature learning techniques in natural language processing where words or phrases from the vocabulary are mapped to vectors of real numbers. Methods to generate this mapping include neural networks, dimensionality reduction on the word co-occurrence matrix, probabilistic models, and explicit representation in terms of the context in which words appear. Word and phrase embeddings, when used as the underlying input representation, have been shown to boost the performance in NLP task such as semantical understanding, syntactic parsing and sentiment analysis.
And in this experiment we use GloVe embedding trained on Wikipedia 2014 plus Gigaword 5 (6B tokens, 400K vocab, uncased, 50-dimension vectors: download here [10]).

4.5.3 *10-fold Cross Validation*

In this part, we will compare decision tree performance with two different feature representation. And compare their accuracy, precision, recall, f1-score. Table 1 summaries the average accuracy and f1-score for these two methods and Figure 8 shows the performance of four categories respectively. We can conclude that with tf-idf outperforms the word embedding in this experiment. Our methods also outperform the method using KL divergence presented in [11]. They achieved the average accuracy of 0.74.

	Avg. accuracy	**Avg. f1**
tf-idf	0.899	0.895
Word embedding	0.795	0.795

Table 1: Average performance Comparison

4.6 Scoring

In this section, we will explain how we find the sentiment score of each attributes and establish their relations with the

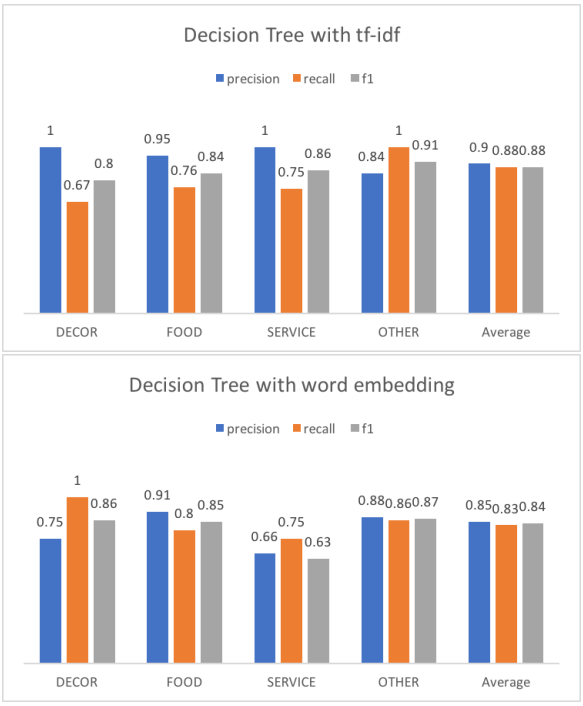

Figure 8: Performance comparison on two ways of feature representation

categories. In Section 4.4, we have classified the extracted nouns into the four categories: DECOR, FOOD, SERVICE and OTHER. Now we will give each category of them a sentiment score. The idea is, we give each extracted attribute a sentiment score by scoring the modifier that specifically indicate reviewers' opinion toward this attribute. For example, for the attribute of "food", we identified the modifiers "excellent" "good" and "so-so", etc. As long as we have the modifiers, we can compute their scores and use an overall score to represent the score of this attribute, i.e. "food" in this case. We used Stanford Parser to find the modifier and noun pair as discussed in Section 4.2.

In order to obtain the sentiment score of each of the modifiers, we used the AFINN sentiment score[12] of each modifier. AFINN is a list of English words rated for sentiment score between -5(negative) to 5(positive). We use this scoring system because it is widely used in sentiment analysis and its newest version contains 2477 words and phrases which cover all modifiers in our data. The AFFIN words are domain independent meaning the modifier has the same score in different context. It is a case that apply to our work because restaurant reviews is part of everyday language and its context is more general than some fields, say, politics. We only count nouns that occur more than five times so make sure that it is a frequent attribute mentioned, or cared, by reviewers. The threshold is set to 5 based on the distribution of frequency of modifiers in the data.

Then we calculate the overall sentiment of the each noun weighed by the times this noun is modified by the modifier as the sum of its AFINN score. For example, when we extracted the attribute "the service", we also extracted the modifiers: "attentive", which occurs three times; "slow",

which occurs one times, etc. So we used the AFINN score of "attentive" times its weight, which is three, and do the same to other modifiers and sum them up. Then we divided the sum by the total weights to get the final score for this attribute "the service". We normalized the sentiment score of each attribute to 1 to 5 so as to match the rating in Yelp and TripAdvisor so that customers have a better idea of how good this attribute is compared with the general rating.

Since we have classified the attributes into different categories, we have the scores of attributes in each category now. For example, under the category of "service", we can see the score of "chef" is 2.7 and the score of "servers" is 2.8, which is not rated as high as its decorations. These scores are calculated by each sentiment score of a modifier times the frequency of its With these scores. The overall performance of each category could be represented by the scores of all the attributes under this category. Figure 9 explains the process of giving sentiment scores.

Figure 9: A sample summary of Table 17 restaurant

5. CONCLUSIONS AND FUTURE WORK

To sum up, our work aims at providing an individual score for each of the four categories in a restaurant's review: DECOR, FOOD, SERVICE and OTHER. We feel the general ratings system most platforms, such as Yelp and TripAdvisor, are using do not tell customers enough information about the different aspects of the restaurant. Our work would be a meaningful supplement. One one hand, our work provides ratings or sentiment scores for each category of each restaurant. On the other hand, our scoring is based on the text reviews customers wrote rather than subjective number rating, which may be misleading because 4 stars may means "good" for one customer but represents "so-so" for another.

Through extracting the modifier-attribute pair in the reviews, we further did feature pruning to find the frequent attributes and classify them into different categories with Decision Tree. We labeled 780 attributes extracted by hand and evaluated the performance of our classifier with two different feature selection techniques: Tf-idf and word embedding. It turns out that Tf-idf outperforms word embedding and reaches the average accuracy of 0.899 and average f1 of 0.895. Our model also outperforms previous work that uses KL divergence, which only reaches the accuracy of 0.74, and f1 score is not mentioned. Lastly, we give each attributes a weighted sentiment score based on AFFIN score and on the frequency of the modifier so we have scores of attributes in each category.

As our project succeeds in providing the ratings of FOOD, DECOR, SERVICE and OTHERS, there is a lot more that can be done based on our work. First of all, attribute extraction can be improved further by integrating people's domain knowledge. We based our extraction and classification on the attributes that occur in the reviews, but we as human beings know that a review on a restaurant could include features such as freshness, waiting time, parking, etc. If this knowledge is integrated, more categories could be labeled and the result will be richer and even more meaningful.

Second, it is possible to extract the name of dishes in a restaurant and give it a score based on reviewers' words toward it. It will give a guidance for customers to find the best dish ad avoid the "not-as-good" dishes in a restaurant. This direction is challenging and promising because the technique to detect name of a dish is not as developed as others. For example, as we did experiment, Google Natural Language API will only mark "Peking Duck" as stuff instead of food. However, this field is promising because integrating human knowledge and do some training on dish names might be a way to solve this problem.
Last but not least, it would be very useful if our work can be transformed into a web/mobile service so that as soon as users post a review, the service will extract the desired features and score them simultaneously. As far as we know, this feature is widely used and accepted by Taobao(www.taobao.com) users in China, but not any English websites implemented this feature. Therefore, these possible directions for future research are very promising. Data Mining in online reviews can surely serve the society in a more efficient and effective way.

6. WORK DISTRIBUTION

Task	Assignment
Proposal	M:Related work, Evaluation, Schedule C:Preprocessing,Feature pruning & Extraction,Sentiment L:Introduction, Dataset, References
Dataset	M &L: Data Visualization &exploration C: Extract restaurant reviews
Processes	C&M&L: Standford Parser
Extraction	C: opinion extraction M&L: attribute extraction
Pruning	C:Obj/subj classification
Classification	C&L:classification& evaluation
Sentiment	M: sentiment analysis & produce summary
Presentation	L:Motivation, Dataset C:Feature extraction & pruning, classification M:Scoring, Future work
Report	M:Scoring, Conclusions and Future work C:Preprocessing, Feature extraction & pruning, classification, Evaluation L:Abstract, Introduction, Related work, Dataset

Table 2: Word distribution with last name initials

7. REFERENCES

[1] Zhang, Ziqiong, Qiang Ye, Zili Zhang, and Yijun Li. *Sentiment classification of Internet restaurant reviews written in Cantonese.*. Expert Systems with Applications 38, no.6: 7674-7682, 2011.

[2] Kang Hanhoon, Seong Joon Yoo, and Dongil Han. *Senti-lexicon and improved NaÃŕve Bayes algorithms for sentiment analysis of restaurant reviews.*. Expert Systems with Applications, 39(5), pp.6000-6010, 2012.

[3] Manek AS, Shenoy PD, Mohan MC, Venugopal KR. *Aspect term extraction for sentiment analysis in large movie reviews using Gini Index feature selection method and SVM classifier*. World wide web, Mar 1;20(2):135-54, 2017.

[4] Jiawei Han, Micheline Kamber, Jian Pei *Data Mining: Concepts and Techniques*. 3rd Edition 2011

[5] Word cloud `https://www.jasondavies.com/wordcloud/`

[6] The Stanford Parser: A statistical parser `https://nlp.stanford.edu/software/lex-parser.shtml`

[7] Partofspeech tags used in the Penn Treebank `https://www.ling.upenn.edu/courses/Fall_2003/ling001/penn_treebank_pos.html`

[8] Subjectivity and sentiment classification using polarity lexicons `https://github.com/nik0spapp/usentc`

[9] Scikit learn: Decision Tree `http://scikit-learn.org/stable/modules/tree.html`

[10] GloVe: Global Vectors for Word Representation `https://nlp.stanford.edu/projects/glove/`

[11] Information extraction over restaurant reviews for the Yelp Dataset Challenge `https://github.com/knowitall/yelp-dataset-challenge`

[12] AFINN `http://www2.imm.dtu.dk/pubdb/views/publication_details.php?id=6010`