CS534 final project milestone
Yaonan Zhong
May 16, 2014

Title: Predicting Star Rating base on Yelp User Review Text

Yelp ratings brings us a new way to choose a business as a customer and run a business as an owner in our daily life. We prefer restaurants or hotels with higher ratings which determine our choice most time. However, we know that not all user ratings are objective all the time. It is possible that a very positive review may come with a five-star rating while another similar one just has a three-star rating. And sometimes we may not have enough time to read all the reviews before making a decision. So can we learn a model to predict the rating for a review text? The underlying goal here is to attenuate the effect of subjective reviews by learning rating from a large number of examples. Note that we can consider subjective ratings as noises in our learning model.

The Yelp dataset we will work on has information on businesses, reviews, users and check-ins. We will focus on all of the restaurant reviews. In learning model selection we plan to use Navie Bayes model, decision tree, and support vector machine as our choice, and we can compare the performances between different models. Since we are doing document classification, one of the important thing is feature selection. How do we construct our feature so as to obtain best performance? That would also be a part of our research. We plan to use k-fold cross validation in our training. We will also estimate the sufficient number of examples for PAC learnability. If the time is allowed, we will try to apply Latent Dirichlet Allocation in topic discovering.

Part 1 Generating Review Text Features

The first task of our learning is generating feature vector for our review text. We can generate a feature vector base on each restaurant or each review, since both of them have their own star ratings (label). First we consider representing each review as a vector including a list of key words. And we need to decide the dimension of the vector and the method of selecting the keywords. The simplest one is adopting the top k frequent words from text reviews of all restaurants as our vocabulary. And then we count the occurrence of each key word in each review. Finally we divide the number of occurrence with the total number of occurrence of all the top k key words to calculate the frequency of our feature vector members, as table 1 shows. A similar way is to calculate the frequency of each key words in all reviews of each restaurant as table 2 shows. And we choose k to be 25, 50, 100, 200 and 500 to obtain least error.

Table 1 Feature generation base on each review

	key word 1	...	key word k	star
review 1	freq of word 1	...	freq of word k	3

...	...	...	...	...
review n	...	...	...	4

Table 2 Feature generation base on each restaurant

	key word 1	...	key word k	star
restaurant 1	freq of word 1	...	freq of word k	3
...	...	...	...	...
restaurant n	...	...	...	4

The review-base feature can be trained and predict star rating for each new unlabeled review, while the restaurant-base one can predict rating for each restaurant. We will choose the restaurant-base one in our following learning step.

In the above method, we select the top k words from a bag of words of all reviews. However, we can filter the words with Part-of-Speech to obtain more meaningful keywords. Here we will choose the top k words from a bag of adjective words of all reviews instead. Words like “awesome”, “bad”, and “delicious” give us more information than “potato”, “chicken” and “I” regarding to the star rating.

Part 2 Selecting Learning Models

Base on the feature vectors generated in part 1, we will apply different learning models to our training samples.

1. Navie-Bayes Classifier

We will implement the multinomial event model with Laplace smoothing.

2. Support Vector Machine

Part 3 Result and Discussion

References