Report for Google local visit prediction and category prediction

Yilun Jin, A15361514

A report submitted for course CSE 158, UC San Diego

November 2017

1 Visit Prediction

1.1 Method Overview

The approach I took was basically trying to predict according to previous user activities and similarities between businesses, via a linear classification model.

1.2 Feature Design

The features representing a data point should have some influence on a user's visit behavior. As far as I consider, I think the following should be represented in the feature design.

1. The general popularity of the business. The higher, the more likely people will visit it.
2. The general activity level of the user. The higher, the more likely the user is going to visit other businesses.
3. The quality of the business. The higher the quality is, the more likely people will visit it.
4. Something measuring the similarity between the user and the business, in other words, some metric that measures user preference. Such metric could be Jaccard similarity of businesses, users, or vectors learned through latent factor model.

Therefore, the feature vector designed is a 21-dimensional vector comprising of the following data. Suppose each piece of datum have the form (u, b), in which u represents the user and b represents the business. And let V_u represent the set of business u have visited, V_b represent the set of users who visited the b.

1. The average rating of b, representing its overall quality.
2. $|V_b|$, representing its overall popularity.
3. $|V_u|$, representing the user's activity level.
4. Jaccardian Similarity and Pearson correlation between b and the user's most visited business, i.e
$$Jaccard, Pearson(b, \arg\max_{i \in V_u} visit\ times(i))$$
representing whether the business fits the user's preference.
5. Jaccardian Similarity and Pearson Correlation between b and the user's highest rated business, i.e.
$$Jaccard, Pearson(b, \arg\max_{i \in V_u} rating(i))$$
.
6. Average Jaccardian Similarity and Pearson Correlation between b and businesses which the user visited, i.e.
$$\frac{1}{|V_u|} \sum_{i \in V_u} Jaccard, Pearson(b, i)$$
7. For all $i \in V_u$, choose three of them which have the highest Jaccardian Similarity and Pearson Correlation with b, and put their Jaccardian Similarity and Pearson Correlation in the feature vector, i.e.
$$\max_{i \in V_u} Jaccard, Pearson(b, i)$$
and repeat the process three times.

8. For all $i \in V_u$, choose the smallest Pearson correlation with b in all $i \in V_u$. i.e

$$\min_{i \in V_u} Pearson(i, b)$$

I did not do it for Jaccardian Similarity because the value I get must be 0.

1.3 Model Selection

I decided to try two models, logistic regression and support vector machine. Finally I chose to use logistic regression for two reasons. First, training a support vector machine is costly. It took three hours to train a SVM on the training set, while it took only several minutes for a logistic regression model to converge. Second, logistic regression outperforms SVM for any parameter C chosen.

1.4 Implementaton

I chose the package TensorFlow to implement my logistic regression for its quickness. Also, in order to reuse some of the code with assignment 1.2, I modified logistic regression a little bit. Instead of maximizing likelihood on training data, I chose to minimize loss, which is measured by cross entropy between label and the output. Namely,