A Machine Learning based Yelp Recommendation System

Pranav Bhardwaj¹, Nicolas Bievre², Frederik J. Melbye³
pranavb@stanford.edu, nbievre@stanford.edu, frederme@stanford.edu
Department of Statistics, Stanford University¹,²
Institute for Computational and Mathematical Engineering, Stanford University³

Abstract

The Yelp Open Challenge data set contains user, business, and review data from 2004 to 2018. Binary classification is performed using user and business features to predict positive labels of 5 star ratings. High predicted probabilities correspond to recommendations. Linear and tree based models were used. Tree based models performed well due in part to their ability to represent nonlinear relationships. Models generalized well to the future.

Data

Each row corresponds to a review containing user and business information. The label y is positive (1) if the user left the business a 5 star review, and negative (0) otherwise.

Table 1: Train-Validation-Test Split

<table>
<thead>
<tr>
<th></th>
<th>Reviews</th>
<th>Users</th>
<th>Businesses</th>
</tr>
</thead>
<tbody>
<tr>
<td>Train</td>
<td>4,903,362</td>
<td>1,259,558</td>
<td>125,159</td>
</tr>
<tr>
<td>Validation</td>
<td>527,276</td>
<td>262,764</td>
<td>75,605</td>
</tr>
<tr>
<td>Test</td>
<td>527,276</td>
<td>274,722</td>
<td>75,254</td>
</tr>
</tbody>
</table>

Features

Input information to learning algorithms, $x$, is a concatenation of the user features, $x_u$, and the business features $x_b$. A single example $i$ is then:

$$x_i = (x_u^{(i)}, x_b^{(i)}) \in \mathbb{R}^d, \quad d \in \{0, 1\}$$

Where $d = 166$ is the number of predictors.

Models

Logistic regression

In logistic regression, the predicted probability of a 5 star rating is of the form:

$$h_\theta(x) = g(\theta^T x) = \frac{1}{1 + e^{-\theta^T x}}$$

Models are trained using $L_1$ and $L_2$ regularization with unit regularization strength.

Gaussian Discriminant Analysis (GDA)

In GDA, it is assumed that $p(x \mid y)$ follows a multivariate normal distribution. Model parameters $\phi, \mu_y, \Sigma$ are fit by maximizing the log-likelihood of the data given by:

$$\ell(\phi, \mu_y, \Sigma) = \log \prod_{i=1}^n p(x^{(i)} \mid y^{(i)}, \mu_y, \Sigma)p(y^{(i)} \mid \phi)$$

A linear decision boundary at which $p(y = 1 \mid x) = 0.5$ is created.

Tree based methods [1]:

Decision Trees: In Decision Trees, the input space $X$ is repeatedly split into two child regions by thresholding a single feature. The work presented uses cross-entropy loss, which is of the form:

$$L_{\text{cross}}(R) = - \sum \hat{y}_c \log \hat{y}_c$$

The predictor split corresponding to the maximum reduction in loss is made at each step.

Random Forests: Random Forest Classifiers are constructed by bagging Decision Tree Classifiers which are trained using a random subset of features. To predict on a new example, the majority vote from the Decision Tree Classifiers is returned.

AdaBoost: AdaBoost classifiers sequentially apply base classifiers $G_m(x)$ (here decision trees) to modified versions of the data. The initial version of the data has uniform weighting. In each successive iteration $m$, observation weights are modified to place more weight on examples misclassified by the previous classifier $G_{m-1}(x)$.

Results

Model selection was based on validation set accuracy, reported in Table 2.

Table 2: Train and validation accuracy, implemented in scikit-learn [2]

<table>
<thead>
<tr>
<th>Model</th>
<th>Train</th>
<th>Validation</th>
</tr>
</thead>
<tbody>
<tr>
<td>$L_2$-regularized Logistic Regression</td>
<td>63.73%</td>
<td>60.53%</td>
</tr>
<tr>
<td>$L_1$-regularized Logistic Regression</td>
<td>74.15%</td>
<td>74.84%</td>
</tr>
<tr>
<td>GDA</td>
<td>74.07%</td>
<td>74.69%</td>
</tr>
<tr>
<td>Decision tree</td>
<td>75.37%</td>
<td>75.97%</td>
</tr>
<tr>
<td>Random Forest</td>
<td>71.58%</td>
<td>71.56%</td>
</tr>
<tr>
<td>AdaBoost</td>
<td>75.33%</td>
<td>76.43%</td>
</tr>
</tbody>
</table>

The final model selected was an AdaBoost classifier with 40 base estimators. Each base estimator was a decision tree classifiers of maximum depth 4. Test set performance is summarized in Figure 1 and Table 3.

Future Directions

- Perform more extensive feature engineering
- Leverage more of the available Yelp data (tips, review text, photos)
- Leverage the community structure in Yelp with collaborative filtering and graph based models
- Provide recommendations to businesses on improvements they can make

References