HARVARD UNIVERSITY

CS281

Final Project Update: Classification for the Yelp Data Set Challenge

Nicolas Drizard & Virgile Audi

December 10, 2015

1 DATA

We are taking part in the Yelp Dataset challenge round 6. We chosed to solve the question of infering categories based on the reviews. The goal is to build a finer way to categorize the Yelp businesses. The global question is the following: How much information can reviews give us on the type of restaurants?

Right now, the users define the label of each entry. This project could be used as an automation tool for Yelp to label properly its data based on the cluster we will find. This would improve the user experience while querying for a specific kind of restaurants.
Data:

- 1.6 million reviews
- 61 000 businesses
- 481 000 attributes (ie hours, parking availabity, take-out, ambience)
- aggregated check-in measurements over time

We focused first on the restaurants and for one city at a time to grasp in a finer way the local relationships of the restaurants. This also makes more sense on from the user's perspective.

2 Approach

Our main goal is to build latent features from the text reviews which would depict the categories of the business. To put it in a nutshell, we would like to build from the text reviews of an entry a vector representation which carries local geometry information. Once we built the document-term matrix containing the words count of the reviews for each business (the reviews are aggregated over each restaurant), we could simply use a matrix factorisation on the counts . To be able to interpret the latent features as topics, we could use a Non-negative matrix factorization (the possible negative features in the SVD cannot stand for cluster assignement). A finer result can be reached with the use of a generative probabilistic model, that's why we chosed the latent dirichlet allocation. The LDA uses a Dirichlet prior on the words distribution over the topics and on the topics distribution over the document which will gives them more freedom than the deterministic approach of the NMF.

3 Latent Dirichlet Allocation

Model

The LDA is a three-level hierarchical Bayesian model. The basic idea that documents are represented as random mixtures over latent topics and each topic is characterized by a distribution over words.

GRAPHICAL MODEL

The generation process for a document $\mathbf{w}$ in a corpus D is the following:

1. Choose the topics representation $\phi \sim \text{Dir}(\beta)$
2. Choose the number of words: $N \sim \text{Poisson}(\xi)$
3. Choose the distribution of topics $\theta_w \sim \text{Dir}(\alpha_w)$
4. For each of the N words:
   a) Choose a topic assignement $z_{n,w} \sim \text{Multinomial}(\theta_w)$
   b) Choose a word $w_n \sim \text{Multinomial}(\phi_{z_{n,w}})$

Parameter Estimation

The main issue relies in computing the posterior distribution of the hidden variables given a document:

$$p(\theta, \mathbf{z}|\mathbf{w}, \alpha, \beta) = \frac{p(\theta, \mathbf{z}, \mathbf{w}|\alpha, \beta)}{p(\mathbf{w}|\alpha, \beta)}$$

This distribution is intractable. There are two main methods to infer it: through a Gibbs sampler or variational inference. In the first case, we are estimating the hyper parameters θ and

ϕ with sample on the different variables. The sampling method will converge to the right model is we sample enough but it takes a lot of time. As a result, we chose the variational inference method which find the variational parameters that optimize a lower bound on the log likelihood. To optimize the bound, we use an expectation maximization method where we estimate the topic distribution for the current document based on its word counts in the E-step and then update the topics distribution in the M-step.

Moreover, to increase the speed of our algorithm based on the idea that we have enough reviews, we chose the online variational inference method. This enables to go over each data point only once. For comparison, on the reviews from the Las Vegas data set, a gibbs sampler provided by python executed in around 50 minutes whereas our online method executed in less than 1 minute for equivalent topics evaluation.

EVALUATION METHOD

We need a measure to evaluate the performance of our model and to tune the hyperparameters. We use perplexity on held-out data as a measure of our model fit. Perplexity is defined as the geometric mean of the log likelihood of the words in the held-out set of docuements given the trained model. In our case, for each document we held out 20% of the words which constitue the test set.

$$perplexity(D_{test}) = \frac{\sum_{d \in D_{test}} logp(words)}{\sum_{d \in D_{test}} |d|}$$

$$perplexity(D_{test}) = \frac{\sum_{d \in D_{test}} \sum_{w \in d} log\left(\sum_{t \in topics} p(w|t) * p(t|d)\right)}{\sum_{d \in D_{test}} |d|}$$

We used this measure to optimize the number of topics K and the hyper parameters of the optimization of the lower bound.

NEXT STEPS

So far, we managed to extract the categories of the restaurants as latent features with a latent dirichlet allocation algorithm on the aggregated reviews for each restaurant.

A future work could be done on the classification part based on these features: on one hand, clustering the restaurants of different cities and comparing the groups with the existing labels to find new categories or to refine the existing one; on the other hand, we could use the features with classifier to predict the existing categories.

As a variant, we could try a supervised LDA, where we use the existing categories as a response variable associated with each document and infer the joint model of the documents and the responses. Lastly, this method is applicable on each review separately to predict the rating of the review, with a supervised lda also to avoid the predominance of the categories in the topics.

4 CLASSIFICATION

Once the feature has been extracted, we can apply supervised learning methods to classify the Yelp restaurants based on their labels on Yelp. Several approaches are possible with regards to the features to solve this multi-labels classigication task

FEATURES

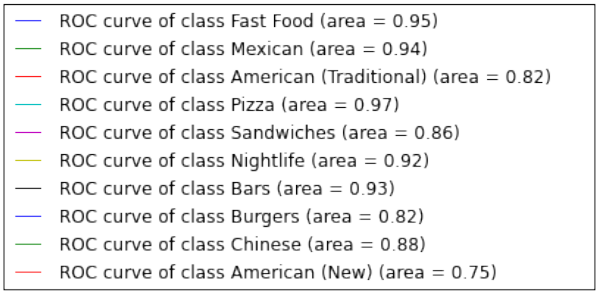
The features are divided into two main parts: on one hand, the topics assignements extracted from the reviews with the LDA and on the other hand, the properties provided by Yelp. Preprocessing the latter part provided a 198 dimensionnal vector. These features contain both numerical features (rating, space coordinates, number of reviews, customer check-in...) and categorical features converted into indicator variables.

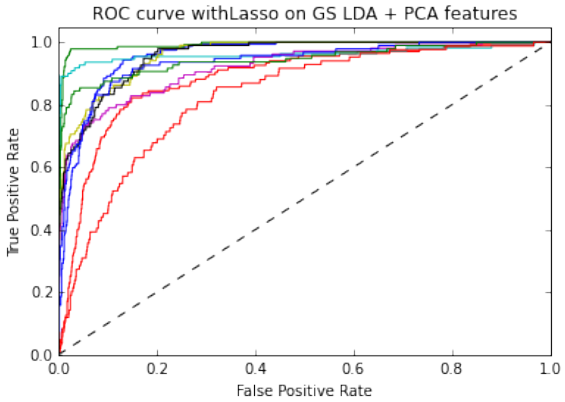
METHODS

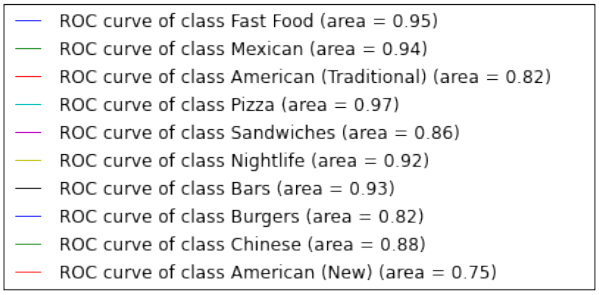
The outputs of our classification are the labels of each restaurant. We chose to train several binary classifiers to predict one label at a time with a One-Vs-All approach. The high dimensionnality of the features and the large number of indicator varibales induces a high sparsity. As a result, a sufficiently robust algorithm need to be used. One particularity is also that the classes are all really skewed, the dummy baseline where we do not predict any label for all the restaurant has a really high score.

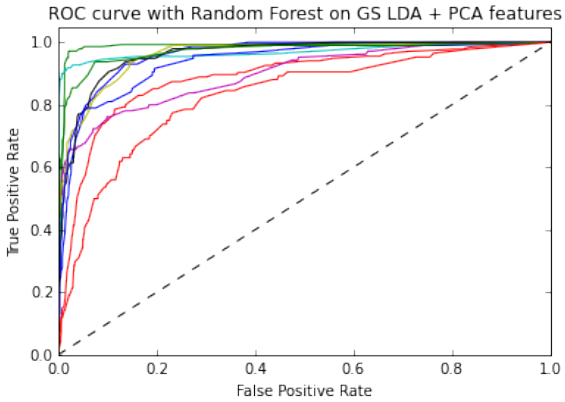
	OVI LDA	GS LDA	NMF
All False	90,02 %	90.02 %	90.02 %
kernel rbf SVM	90,81%	90.38%	90.38%
kernel rbf SVM + PCA	90.83%	90.60%	90.57%
Logistic Regression	90.84%	91.89%	91.24%
Logistic Regression + PCA	92.59%	92.53%	92.68%
Lasso	90.82%	93.44%	92.01%
Lasso + PCA	93.44%	**94.46** %	93.08%
Random Forest	91.90%	93.31%	92.89%
Random Forest + PCA	93.63%	94.04%	93.44%

We first coded a l2 regularized logistic regression and use scikit learn for the other methods. The random forest provides the best predictor expected as it corresponds to an ensemble method that combines multiple decision trees, leading to a more robust model. We showed the ROC curves that we obtained. The variance of the classifiers with regards to the class used as output is quite high. This could be explained by the relative skewness of each class and also their own specificity. Moreover, using dimensionality reduction method on the Yelp check-ins information before the classificationthe PCA also improved the accuracy. These features contains the average number of customers for each hour of the week which leads to a vector of dimension 168. The principal component analysis dropped it down to 13, while keeping 94% of the variance.


ROC curve of class Fast Food (area = 0.95)
ROC curve of class Mexican (area = 0.94)
ROC curve of class American (Traditional) (area = 0.82)
ROC curve of class Pizza (area = 0.97)
ROC curve of class Sandwiches (area = 0.86)
ROC curve of class Nightlife (area = 0.92)
ROC curve of class Bars (area = 0.93)
ROC curve of class Burgers (area = 0.82)
ROC curve of class Chinese (area = 0.88)
ROC curve of class American (New) (area = 0.75)


ROC curve withLasso on GS LDA + PCA features
True Positive Rate
False Positive Rate
0.0
0.2
0.4
0.6
0.8
1.0


ROC curve of class Fast Food (area = 0.95)
ROC curve of class Mexican (area = 0.94)
ROC curve of class American (Traditional) (area = 0.82)
ROC curve of class Pizza (area = 0.97)
ROC curve of class Sandwiches (area = 0.86)
ROC curve of class Nightlife (area = 0.92)
ROC curve of class Bars (area = 0.93)
ROC curve of class Burgers (area = 0.82)
ROC curve of class Chinese (area = 0.88)
ROC curve of class American (New) (area = 0.75)


ROC curve with Random Forest on GS LDA + PCA features
True Positive Rate
False Positive Rate
0.0
0.2
0.4
0.6
0.8
1.0

Moreover, this method provides a way to sort the features by importance.

FEATURES IMPORTANCE