HARVARD UNIVERSITY

CS281

Final Project Update: Classification for the Yelp Data Set Challenge

Nicolas Drizard & Virgile Audi

December 2, 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 **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.

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

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

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