Which beer will you drink next?

Group 22: KaWing Ho, Ankur Sarin, Mert Sanver

Abstract

Previous studies on user-product review data have focused on inferring a user's preferences based on their past reviews and reviews for similar iterms by other users. The objective was to reccommend items that a user might like. Very few studies have considered the chronological order of a user's reviews. For this project, we evaluate how a user's choices can be predicted based on thier past reviews.

1 Introduction

Online product review sites such as Yelp.com are getting increasingly popular and attracting millions of users around the world. The proliferation of product review sites enables the extraction of meaningful insights about consumer behavior which was previously not possible due to the lack of such data. By modeling consumer behavior and choice, companies can better predict the demand for products and orchestrate more targeted marketing efforts. Our project explores how users choose which product to review next based on their past history, and how this choice can be modelled.

While most previous work related to product reviews focused on recommendation techniques such as collaborative filtering, our project focuses on studying a user's choice at each time step. The former aims at predicting the preferences of users, i.e. which items a user might like, whereas we aim to predict their next subsequent choice. Our project is similar to some previous studies on web page prediction where the goal is to predict the next web page a user might visit given their history.

Using data from beeradvocate.com, we try to predict a user's next choice of beer. We extract and evaluate various features that can be used to predict a user's choice and using various models, show that it can achieve 4 times better accuracy than a random baseline prediction.

2 Related Work

[5] presents an approach for using markov models to predict user behavior when navigating websites on the world wide web. Using log data from websites, they attempt to predict the next website that a user may visit. They used the longest repeating subsequences (LRS) to reduce the complexity of their k-th order markov model, while retaining the predictive power. [1] builds on the ideas above and uses a hybrid ANN and Markov model for web page prediction.

[7] and [8] present heuristics that can be used to predict a user's navigation through the wikipedia graph. They also propose a model that can be trained as an automatic agent to navigate through the graph using the same information available to humans. This suggests that we can use simple features to model user behavior. We plan to use similar heuristics and extract features to propose and train a model to predict which beer a user will review next. The problem we tackle is slightly different from the papers above. In our case, a user is not trying to reach a target, like a specific wikipedia page, but instead trying to choose which beer to try next, based on some preferences.

3 Data Collection

The dataset was obtained by crawling www.beeradvocate.com. It is an independent community of beer enthusiasts and industry professionals who are dedicated to supporting and promoting beer. For each review we have a *beer style*, *timestamp*, *textual description* and *rating* which is a weighted sum of various aspects of the beer such as taste, feel, smell and look. In addition, we also get details about each beer such as its *type* and *brewery*. Examples of beer styles are American Pale Lager, American Pale Ale (APA), American Double /Imperial Stout, etc. Each beer style can be further categorized into more generic *types* such as American Ales, American Lagers, English Ales, German Lagers etc. There are a total of **104** styles and **14** types of beer. The dataset also includes the *location* of the each reviewer. Below are some summary statistics:

Number of users: 23,548
Number of beers: 28,866
Number of reviews: 625,032
Number of Locations: 9,094
Number of Breweries: 5840
Most popular type: American Ale
Most popular styles: American and Imperial IPA
Most common user location: Chicago

4 Dataset Exploration

We explored the dataset to evaluate which attributes from a user's past reviews provide information about which beer styles users choose to review in the future. For example, if a user reviewed a particular sequence of styles or types, would they be more likely to evaluate certain styles than others? We define a set of attributes as X_j where $X_j = \{style, type, breweryid, location, etc\}$. We also define a window of size k for an attribute X_j as the values of the attribute X_j in k consecutive reviews $\{x_{j_1}, ..., x_{j_k}\}$. For example a window of size 3 for the attribute *brewery id* could be {*Samuel Adams, Heineken, Guinness*}.
Given the past reviews of a user, we evalutaed the mutual information between different attributes (X_j) in their previous k reviews and the beer style (Y) they choose to review next. Figure 1 shows the mutual information for window sizes $k = 1...10$:

$$I(X;Y) = \sum_{x \in X} \sum_{y \in Y} p(x,y) log(\frac{p(x,y)}{p(x)p(y)})$$

Mutual information calculates the mutual dependence between the variables. If knowing the values of an attribute in the k previous reviews does not provide information about which beer style might be reviewed next, i.e. they are independent, the mutual information will be 0.

As shown in Figure 1, the mutual information is non-zero even for small k for certain attributes. Even though the mutual information increases as the size of the window k increases, the prior probability for each k-length sequence also drops exponentially. Intuitively, this is expected as the number of distinct windows increases exponentially with k. This effect is explored more in section 6.

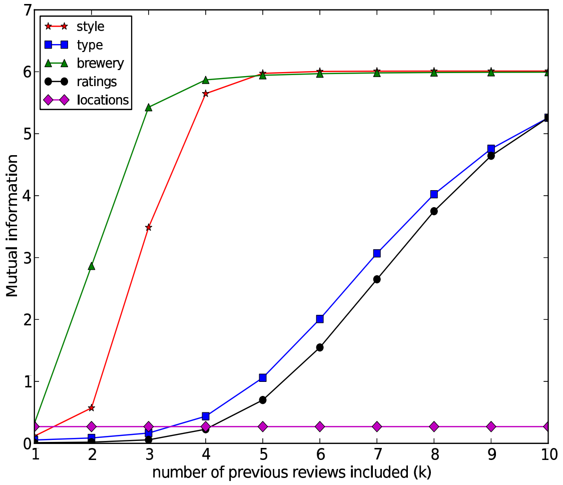

Figure 1: Mutual information between reviewed beer's styles and previous reviews' features

We also noticed that users tend to review similar beers multiple times in a row. Similar beers could mean beers of the same type, style or from the same brewery. To measure this effect, we computed the probability for a user reviewing k consecutive beers with the same value of an attribute over different window sizes $k = \{2..10\}$. In other words, $P(x_{j_k} \mid x_{j_0}, x_{j_1}, ...x_{j_{k-1}})$ where $x_{j_0} = x_{j_1} = ... = x_{j_k}$, for a particular attribute X_j. Figure 2 shows the comparison between the dataset and a random baseline generated through a Monte Carlo simulation based on the prior distribution of each attribute. As can bee seen in the figure, users tend to repeat the same type, style and brewery than expected using the random baseline. For example for the attribute *type* and window size 3, we see 14% repetition compared to 8% for the random baseline. The results are similar for other attributes as well.

4.1 Changes of reviews over time

Another interesting aspect to explore is how a user's review pattern changes over time. Do they review more distinct or similar beers over time? To evaluate this, we divided up the reviews into time windows, each of size $k = 10$ and measured the number of distinct styles, types and breweries within each time window. Figure 4 shows the mean and standard deviation of distinct items

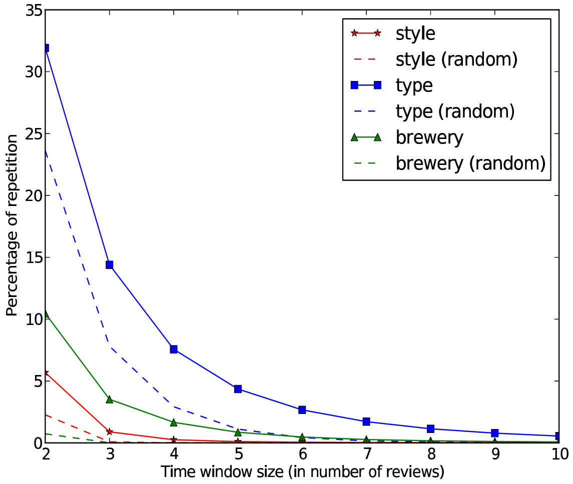

Figure 2: Probability of reviewing similar beers by attribute

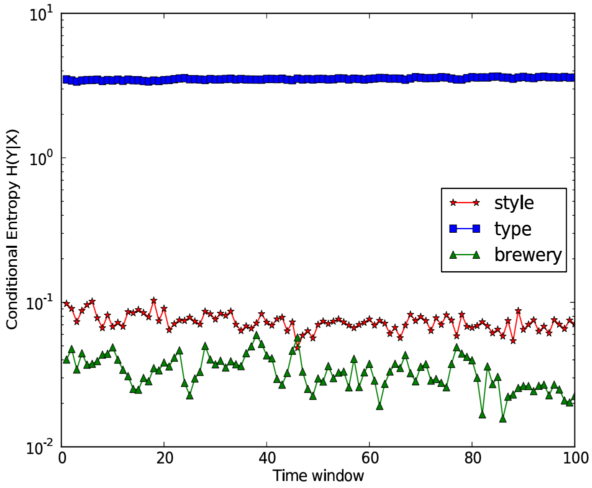

Figure 3: Conditional Entropy

in each time window for each attribute. Similarly, we computed the conditional entropy of overlapping time windows of size $k = 3$ between different attributes in past reviews and the subsequent beer style chosen (Figure 3):

$$\begin{aligned} H(Y|X) &= H(Y|X_{j_1}..X_{j_k}) \\ &= -\sum_{y \in Y} p(y, x_{j_1}, ...x_{j_k}) log(p(y|x_{j_1}...x_{j_k}) \end{aligned}$$

Figure 4 shows that for all attributes, the number of distinct values for the attributes do not change over time and their variance is small. Thus we can conclude that all users maintain a similar review pattern over time i.e. they do not start reviewing more distinct or similar beers over time. Figure 3 also suggests that the predictability of the beer style to be reviewed next based on past reviews doesn't change much over time. Hence we can consider the choice for the next beer as time independent.

5 Models for prediction

As shown in the previous section, past reviews of users can provide hints for which beer style a user may review in the future. We attempt to predict a user's choice using various models:

- K-th order Markov model
- Naive Bayes model
- Softmax regression
- Mult-class SVM

5.1 Features selection

As discussed in section 4, each attribute for a user's past reviews can provide varying degrees of information for a user's next choice. For each model described in this section, we used the following attributes from the k previous reviews as features:

- user's location
- beer style
- beer type
- brewery
- review rating

While it is possible to represent features as bit vectors for the k previous reviews, the dimension for some attributes such as location is too high (9,094). To reduce noise and possibility of overfitting, we selected only the top 20 locations with the highest mutual information. For the ease of computation, we also divided the ratings into discrete intervals of size 0.5.

5.2 K-th order Markov model

In [4, 5], the authors tried to model the probability of a user traversing a web page given a set of web pages they visited previously. In particular, they considered

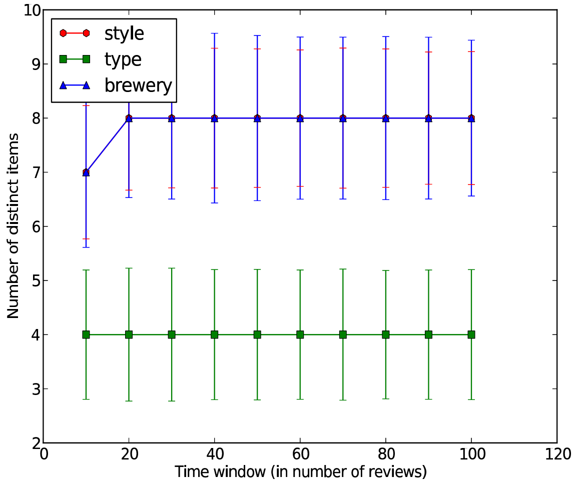

Figure 4: Distinct number of items

the last k pages a user visited as an n-gram and tried modelling it as a k-th order Markov model. Similarly, for a given review, let's denote the beer style reviewed as $Y^{(i)}$. Considering the k reviews prior to it, we can denote the values of an attribute X_j in those k previous review as $X_{j_1}^{(i)}, ..., X_{j_k}^{(i)}$ respectively. Similarly, we can construct a k-th order Markov model to predict a user's choice of beer($Y^{(i)}$) given the values of an attributeX_j in k previous reviews($X_{j_1}^{(i)}, ..., X_{j_k}^{(i)}$. The label is predicted as per:

$$\arg\max_y P(Y = y \mid X_{j_1}, ..., X_{j_k})$$

X_j are the features described in section 5.1 and k ranges from 1 to 10. The probability $P(Y = y \mid X_{j_1}, ..., X_{j_k})$ was estimated using maximum likelihood estimation for the parameters where the log likelihood is given by:

$$L(\theta) = \sum_{i=1}^{m} P(y^{i}) \mid x^{(i)}; \theta$$

Thus,

$$P(Y = y \mid X_{j_1}, ..., X_{j_k}) =$$

$$\frac{\sum_{i=1}^{m} 1\{Y^{(i)} = y, X_{j_1}^{(i)}, ..., X_{j_k}^{(i)}\}}{\sum_{i=1}^{m} 1\{X_{j_1}^{(i)}, ..., X_{j_k}^{(i)}\}}$$

To determine the optimal window size k for each attribute, we evaluated accuracy for $k = 1...10$ and picked the k that gives the least error. The optimal window sizes for different attributes are discussed in section 6.

5.3 Naive Bayes model

We also tried using a generative model to predict a user's choice for the beer style. We tried a simple Naive Bayes model [3]. In essence, the model predicts the beer style y to be reviewed by a user in the future as:

$$\arg\max_y P(Y = y) \prod_{j=1}^{4} P(X_{j_1}, ..., X_{j_k} \mid Y = y)$$

$P(X_{j_1}, ..., X_{j_k} \mid Y = y)$ is estimated from the data set with Laplacian smoothing:

$$\frac{\sum_{i=1}^{m} 1\{Y^{(i)} = y, X_{j_1}^{(i)}, ..., X_{j_k}^{(i)}\} + 1}{\sum_{i=1}^{m} 1\{Y^{(i)} = y\} + |X_j|^k}$$

Again, X_j correspond to different attributes including *styles*, *types*, *brewery* and *location*. We again searched for different values of k for each attribute.

5.4 Softmax Regression

The Softmax Regression model generalizes the logistic regression model for a multi-class classification problem. The label is picked as per:

$$\arg\max_j P(y^{(i)} = j \mid x^{(i)}; \theta)$$

where,

$$P(y^{(i)} = j \mid x^{(i)}; \theta) = \frac{e^{\theta_j^T x^{(i)}}}{\sum_{i=1}^{l} e^{\theta_l^T x^{(i)}}}$$

We use maximum likelihood estimation to evauluate our parameters where the log likelihood is given by:

$$L(\theta) = \arg\max_\theta \sum_{i=1}^{m} \sum_{j=1}^{k} 1\{y^{(i)} = j\} log \frac{e^{\theta_j^T x^{(i)}}}{\sum_{i=1}^{l} e^{\theta_l^T x^{(i)}}}$$

The log likelihood is maximized using stochastic gradient ascent where the gradient is given by:

$$\nabla_{\theta_j} L(\theta) = x^{(i)}(1\{y^{(i)} = j\} - p(y^{(i)} = j \mid x^{(i)}; \theta))$$

5.5 SVM

SVM is one of the best off the shelf classifiers. We use the LIBSVM Library implementation of SVM [2] with a linear kernel to train our model. It uses a ”one-against-one” approach [6] for multi-class classification. For k classes, we construct and train $\frac{k(k-1)}{2}$ classifiers for each combination of the classes. The label is picked using a ”voting” strategy. The output of each binary classifier is considered to be a vote and the class with the maximum number of votes is picked as the label for a data point. The parameters for the model are picked using cross validation.

6 Evaluation

6.1 Training and Test Set Generation

To train the models mentioned in section 5, we preprocessed the data to generate features based on the window size k. Each data point was a group of $k+1$ consecutive reviews, using attributes from the first k reviews to generate the features and the beer style in the $k+1$-th review as the label. We only included users with at least 100 reviews and only considered their first 100 reviews to train and test our models, since the review patterns were found to be time independent. These users were randomly split 80%-20% into test and training sets.
Training set size:**2526** users
Test set size: **631**

6.2 Evaluation metrics

We used **classification accuracy** and **mean reciprocal rank** as evaluation metrics. Classification accuracy:

$$\frac{\sum_{i=1}^{m} g(X_j^{(i)};\theta) = Y^{(i)}}{m}$$

where $g(X_j^{(i)};\theta)$ is beer style with the highest probability predicted by the model and m is the number of test samples. In addition to just predicting the most probable beer style for a given test data point, we also did a **top n prediction** by considering the 5 and 10 most probable labels predicted by each model. If the actual beer style $Y^{(i)}$ is among the top 5 or 10 prediction, it was considered a correct classification for this metric. Mean Reciprocal Rank:

$$\frac{\sum_{i=1}^{m} rr(g(X_j^{(i)};\theta))}{m}$$

where $rr(g(X_j^{(i)};\theta))$ is the reciprocal of the relative position of the actual beer style in the top n predicted labels. For example, if the actual beer style is ranked third by a model, then the reciprocal rank is 1/3. The mean reciprocal rank is a fairer metrics for top n prediction as it also takes the “ranking“ into account. To evaluate the performance of the models described in section 5, we compared their accuracy using the metrics described above against a random baseline model. The random baseline classifies data points by randomly picking from the prior distribution for each beer style.

6.3 Results and Discussion

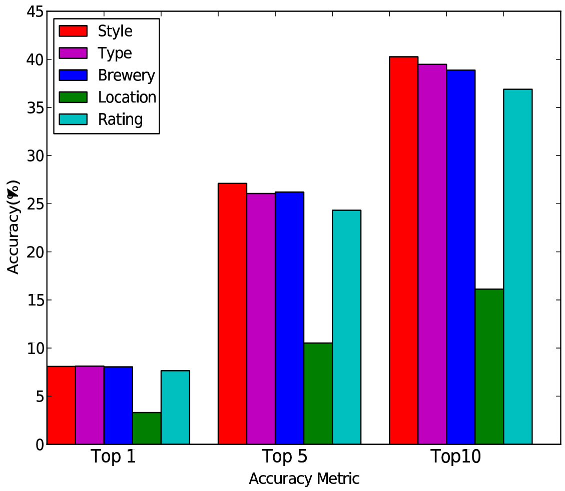

Figure 5: The accuracy of different attributes with k-th order Markov model

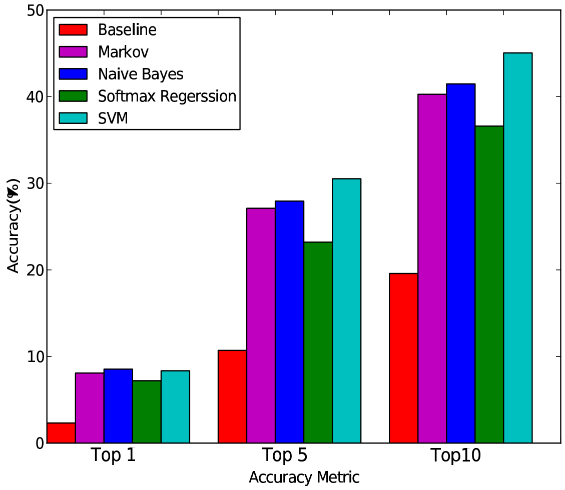

Figure 6: The accuracy of different models

Figure 5 shows the accuracy of the k-th order model Markov model for each attribute. We observe that *style* is the most useful attribute for predicting a class label. It’s closely followed by *type* and *brewery*. As

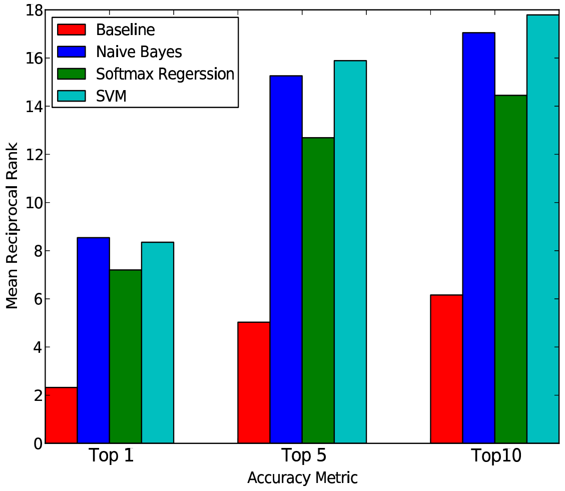

Figure 7: The mean reciprocal rank of different models

mentioned in section 5.2, the optimal window size k for each attribute is chosen by searching for $k = 1...10$. The optimal values of k for *style*, *type* and *brewery* are 1, 2 and 1 respectively. As mentioned in section 4, as k increases, the prior probability of a particular instance of k-sized window decreases exponentially, resulting in smaller predictive accuracy as most instances of the k-sized windows are never observed in the training set.

Figure 6 shows the classification accuracy of various models using the top 1, top 5 and top 10 metrics. Overall, SVM performs the best and it's closely followed by Naive Bayes. For top 1, both Naive Bayes and SVM produce accuracy beyond 8% while the random baseline has accuracy of about 2.3%. While all models perform better than the random baseline, softmax regression lags behind other models. Given the high dimension of the feature vectors (even after feature selection described in section 5.1), softmax regression is likely suffering from the curse of dimensionality. As SVM is a max margin classifier, it performs much better with high dimensional data. It's surprising that Naive Bayes does so well compared to other models given that the assumption of conditional independence between features such as *styles* and *brewery* isn't always correct.

Figure 7 shows the mean reciprocal rank of various models. Unsurprisingly, the scores are lower than those evaluated by the simple classification accuracy metric. However it is interesting to note that for the top 5 prediction, most of the models score about half as much as the simple accuracy metric. This implies that on average, the correct label is ranked second. Another interesting thing to note is that there isn't much increase in the score when going from top 5 to top 10 metrics. This implies that if we in fact make a correct prediction for a label, it is ranked high, otherwise we miss it altogether.

As shown in the results above, our models can do up to 4 times better than random prediction. This implies that we can model how users choose which beer they might try next using hints from past reviews. It also shows that the process for choosing the next beer may not be completely user specific as we can use the choices learned from a set of users to predict the behavior of other users. There is a key distinction between our work and recommendation techniques such as collaborative filtering. These models try to predict a user's preference for each item. On the other hand, our approach focuses on how to predict a user's future behavior. We focus on how a user might choose a certain item, in our case beer, based on their history and behaviour of other users in similar situations. We show that it is possible to extract useful features and model a user's choices.

7 Future Work

We explored how various attributes such as beer styles, types and breweries of a user's past reviews provide information about a user's future choice of beer. However we did not consider the actual textual description of past reviews. It may be interesting to also incorporate these as features for our models in the future. In addition, we noted the major distinction between recommendation techniques such as collaborative filtering and our work. The former focuses on predicting items that a user might like, by predicting their preferences, whereas we focus on modelling their next subsequent choice. It would be interesting to quantitatively compare and contrast our results against such a recommendation model like collaborative filtering.

References

[1] M. A. Awad and J. L. R. Khan. Web navigation prediction using multiple evidence combination and domain knowledge. *IEEE Transactions on Systems, Man, and Cybernetics*, 37:1054–1062, 2007.

[2] C.-C. Chang and C.-J. Lin. LIBSVM: A library

for support vector machines. *ACM Transactions on Intelligent Systems and Technology*, 2:27:1–27:27, 2011. Software available at `http://www.csie.ntu.edu.tw/~cjlin/libsvm`.

[3] A. Mccallum and K. Nigam. A comparison of event models for naive bayes text classification, 1998.

[4] P. Pirolli, P. L. T. Pirolli, and J. E. Pitkow. Distributions of surfers' paths through the world wide web: Empirical characterizations. *World Wide Web*, 2:29–45, 1999.

[5] J. Pitkow and P. Pirolli. Mining longest repeating subsequences to predict world wide web surfing. pages 139–150, 1999.

[6] G. D. S Knerr, L Personnaz. Single-layer learning revisited: A stepwise procedure for building and training a neural network, 1990.

[7] R. West and J. Leskovec. Automatic vs human naviagtion in information netowrks. *ICWSM*, 2012.

[8] R. West and J. Leskovec. Human wayfinding in information netowrks. *WWW*, 2012.

Some interesting links about beer:

http://daphne.palomar.edu/mlane/BEER/BeerPeriodicTable.jpg

http://blog.beerjobber.com/wp-content/uploads/2012/10/Picture-4.png