Collaborative Dining:  
A Social Recommender System for Restaurants

Hamilton Nguyen [hamiltonnguyen@berkeley.edu, 3 units] and Sylvia Lin [sylviarlin@gmail.com, 3 units]

Abstract

The recommender system is one of the most profound results of our information age. Among the most widespread uses of our vast online communities are consumption, personalization, and discovery. Recommender systems sit perfectly at the center of all three, providing the ultimate service to the consumer by telling them what they want to consume. At the same time, restaurants are an ideal domain for a recommender system. Consumers have an endless demand for restaurants, consumers have highly nuanced and personal opinions of different restaurants, and consumers are always searching for new restaurants to experience. However, designing a recommender system for restaurants is not a solved problem. Current implementations of such systems, despite being considered part of the social networking landscape, do not make use social contexts in their application. The aim of this project is to provide a more robust solution to recommending restaurants. Our proposed solution builds on existing approaches by aggregating information across various forms of online social interactions. By expanding the parameter space of current restaurant recommendation, we aim to provide increased personalization to users.

Introduction

In today’s world, there are various search engines that, given some keywords, can bring up a myriad of page results to choose from. These page results are ranked accordingly based on different algorithms. Regardless of the search engine, the pages will be ranked in the same way no matter how many times a user selects the refresh button. While this interface works for general searches, some situations call for a more personalized search.

When people want to eat out, the most common question that comes up is “where?”. This indecision comes from indifference to the choices at hand; people get stuck going to the same restaurants, which reduces the novelty of any given location. Making a decision then becomes more difficult because no restaurant choice stands out, and a person cannot act on a preference. By creating a restaurant recommender, we can give a user an option that now stands out. There is a factor of randomization and personalization, so the user will not receive the same recommendation in case a previous choice was unappealing.

Recommender Systems and Restaurants

Recommender systems, at least conceptually, are actually relatively old compared to other products of the Web 2.0 revolution. The first recommender system is credited to a team at Xerox PARC that developed the Information Tapestry [1], which filtered newsgroup posts by interest. The first commercial use of recommender system was a service called Firefly, which recommended music to consumers. Nowadays, the application of recommender system is widespread, from marketplace sites such as Amazon and eBay, to content streaming sites such as Pandora and YouTube. There is even a notion of the recommender industry [2], as businesses begin to perceive recommender systems as the more finessed answer to Web advertising.
Presently, the web has a number of search engines or recommendation generators specific to restaurants. Both have their own drawbacks, however. Current search engines produce a lot of data, which is then sorted according to the engine-specific algorithm. However, these engines always produce the same results, so if a user is dissatisfied, refreshing the page will not make anything new turn up. Furthermore, the lack of personalization means search engines cannot recommend well. Finally, search engines cannot truly substitute as recommenders because search engines can only help users find what they know they are looking for. Recommender systems enable discovery, which can be quite rewarding to many users.

On the other end, current recommender sites have algorithms specific to returning personalized results. These sites draw from information within a user base though. Thus, sites with a smaller user base will have less opinions to work with and lack the data required to make a good recommendation. To create a good medium between these two types of sites, we want to draw from the data available within search engines, as well as user feedback from recommendation sites.

Existing recommender systems also have the previously mentioned shortcoming that they do not consider social contexts. To elaborate, consider the following scenario: two users, living in nearly the same location, have almost the same favorite restaurants and have rated restaurants nearly identically. However, the two users are in completely different social circles, and have no friends in common. Current systems would likely give the same recommendations to both users— but is this realistic? Dining, like many leisure activities, is a social experience. Often, people seek out dining options because they are interacting with other people in their social network—and restaurant recommenders must take this into account.

**Design and Algorithm**

Across various literature, there exist a number of categories of algorithmic approaches to solving the recommender system problem. We will examine two of them: content-based filtering, and collaborative filtering [3].

In content-based filtering, the first step is to build a user profile containing items (or, in our case, restaurants) of interest to the user, where ‘of interest’ can mean previously rated, purchased, etc. Every item has associated features or attributes, and the combination of features and attributes of the items in the user profile can be used to characterize the user’s interests. Other items can then be compared with this characterization, via their own extracted features and attributes. Recommendations are made if these other items are seen to “fit” well with the user profile.

Unlike content-based filtering, collaborative filtering looks past the single user, and examines the community as a whole. Rather than matching items with users, user profiles are matched against other user profiles. If two profiles have high similarity, we can predict that items of high interest to one user will be of high interest to the other. We can then make recommendations for items that exist in one of the profiles, but not the other.

At the heart of both approaches is the \( k \) Nearest Neighbors algorithm, which we will be applying extensively. Given a set of vectors and a test vector, the \( kNN \) algorithm will identify which vectors in the set are ‘nearest’ to (i.e. most similar to) the test vector. We can make comparisons between items and profiles or profiles and profiles by extracting relevant information from each, using the extracted information as a vector, and measuring similarity via \( kNN \).

In our application, we will use a hybrid approach that combines both ideas. On their own,
each approach has an incomplete view of the landscape: content-based filtering ignores the existence of other users, while collaborative-filtering does not take into account individual items. Both can contain valuable information needed to make a good recommendation. In addition, we will be integrating social contexts into our considerations. User profiles can contain not only items of interest to them, but items of high interest to the rest of their social network. Likewise, two user profiles can have increased similarity by virtue of being in the same social network. The idea is that users may have interest in items that they would not normally show interest in, if it happens that people they know have interest in those items.

Previous restaurant recommenders were web services for which users signed up. The recommenders were limited in that their data mostly came from the users’ activity while on the web site. For our data set, we will be aggregating information from a variety of social media services. Among the sites that we will crawl are: Yelp, for users’ restaurant history, price ranges, food types, and location; Facebook, for users’ social networks and 'Liked' pages; and Foursquare and other location-based social media for information on where users have been recently. By leveraging the vast quantity of data already available, we do not need to have user subscriptions to our own service in order to do our analysis.

Development Plan

Our current goal is to create an effective restaurant recommendation generator that offers choices similar to a user’s tastes, environment, and budget. By the end of September, we want to be able to crawl websites and extract pertinent information efficiently. We will also start looking into ways to analyze and sort this information. By the end of October, we should have a simple user interface set up, with a simplified algorithm working to produce decent recommendations.

Summary

Restaurants are a well-suited subject to design recommender systems around. Although current implementations exist, they are limited by the size of their user base and the scope of their algorithms. Our application will address these issues in two ways: first, rather than pulling data only from user activity while subscribed to our service, we will consolidate existing information from various social media sites with multiple types of data; second, we will integrate social contexts into our recommender algorithms, drawing additional information from activity in users’ social networks. The goal is a restaurant recommender system that can offer a greater level of personalization.

References

1. Kalseth, Fredrik: “Developing a Restaurant Recommender System” UC Santa Cruz, May 2005
