Vision

Overview

The core functionality of our web app revolves around recommending various activities for people. Using machine learning algorithms, and data collected through a variety of sources, the app will tailor suggestions to match more appropriately. Social features will suggest friends likely to enjoy recommended activities, encouraging group socialization.

Activities include, but are not limited to:

- Entertainment (such as movies, or music)
- Leisure (such as reading)
- Sports (such as jogging, or mini-golf, or soccer)
- Food (restaurant suggestions)

Sources for the activities will range from foursquare, to yelp, to google places, eventful.com, and others.

Signup

New users create an account with our service (or potentially tied to third-party services such as foursquare). Following creation of the account, they will have the opportunity to take a quick interest survey, to help tailor initial suggestions.

Matching

Additional to survey information, preference data may be obtained via foursquare, if possible. Users will also be able to rate suggestions, in a fashion similar to netflix, in order to better improve recommendations. Good matches will be sortable by various aspects including cost, and distance, which may also be factored into algorithms explicitly.

Socialization

Users will be able to connect with friends via their account (either via foursquare, or a simple adding system similar to snapchat). When a user is connected with friends, suggestions will include information about which friends are likely to enjoy those activities with them). Depending on development time, an additional option for the app is to allow users to specify friends, and find an activity likely to be enjoyed by all.

Other

Our primary demographic and main focus would be other college students, but this could be potentially modified to be usable by anyone. It solves the problem boredom. For many college students in an area that they aren't used to they may just not know what's around. The are alternatives to specific parts of the app such as urbanspoon, but nothing that does everything. This is compelling because in the information age there should never be a moment where someone doesn't know something. It's therefore unacceptable that people sometimes don't know what to do. This product will be unique because of it's holistic approach to the recommendation system.

Software Architecture

Application Architecture

Initially this will be developed as a web application, although we intend to develop in an API-centric fashion, so as to allow expansion into native mobile applications if time permits, or post-course if we want. We plan to design the interface to be responsive, and mobile friendly.

Technologies and Frameworks

Frontend interface: HTML, CSS (including responsive aspects), Javascript
Server-side: Our current plan is to implement the website in Ruby, using the Ruby on Rails web server framework in a relatively basic fashion, which will allow easy interconnection with the machine learning algorithms.
Machine Learning: Currently, we intend to implement the Machine Learning algorithms in Ruby, because Ruby provides ways to communicate to servers and as a scripting language it is easy to use. Only concern is if number of users and amount of information becomes too huge the time it take may be longer than desired, at which point we might start to explore R a bit.

3rd Party APIs

A large amount of data will be required to be able to make any recommendations. The Yelp API will be useful for getting large amounts of data on local restaurants. The foursquare API will be useful in both getting user information and information on nearby POI (points of interest). There are many other API's that we can use to find other POI such as eventful and google places.

Challenges and Risks

The largest risk is feature creep. If too many features are started early, that could cause half-finished or sloppy features in the final product. The risk can be minimized by defining what is absolutely necessary for the bare bones product and making sure that gets completed before anything else gets worked on. Unfortunately, there isn't much that can be done and this will be a problem that every group will need to keep in mind.
One challenge is scalability. As number of user and feature, amount of information increases, the app may take too long to respond. This is a common problem with a lot of apps and we'll just have to adjust accordingly when we hit the roadblock.