Restaurant Recommender System

Authors: Unni Nair Ane Unzueta Unai Elizalde

- **Concept:** the user asks for a set of features in a restaurant and the recommender system makes a first approach. The user then gets to further specify from a set of attributes given by the system. Finally, a list of restaurants matching the user's desires is shown.

- **Preparation:** we first got a database with a list of restaurant from different cities across USA. Each restaurant has a list of attribute ID's, and there is another file that links each ID with the specific attribute (257 total attributes). The attributes are not ordered, and most of them belong to different categories, so the dataset is not clean and calculating similarities between restaurants is challenging.
  We decided to work with Chicago, as it was the most complete city. We made a parser that would link each attribute to every restaurant that contains it, so the user would write some keywords that would be used to get a list of restaurants. Then we would use some algorithms to filter out most restaurants a give the user a final list.

- **Implementation:**
  - **Parser:** the parser first reads the list of features from a file, linking each feature to an ID, and saves them in a map. It then reads the list of restaurants from each city: for each, it links every attribute found inside to that restaurant. The result is that each restaurant has a vector of attributes, and each attribute has a vector the restaurants it is featured in.

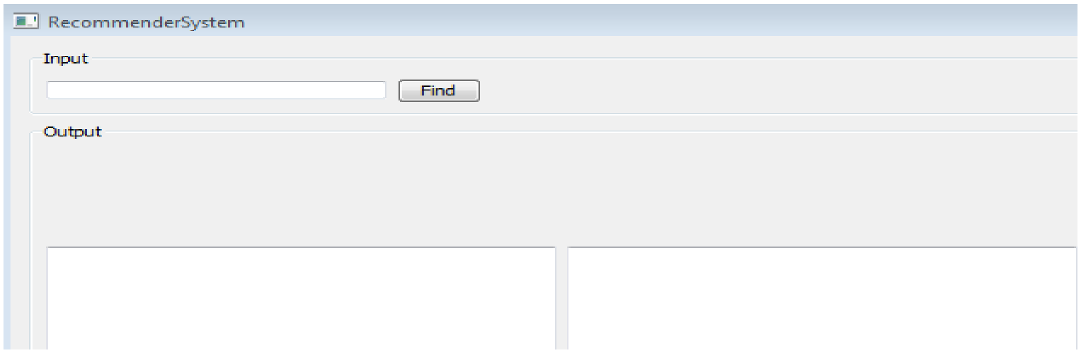
  - **Database:**
    - *Features*: a basic struct containing a name, an ID, a type and a vector of pointers to restaurants that have that feature. Feature types are used for the 4 special cases of features that are mutually exclusive (decor quality, food quality, price range and service quality).

    - *Feature Database*: a class with a map of features that uses the feature name as the key. It also has a vector with all the feature names. This class implements the functions that allow the system to use keywords to search for restaurants: feeding it feature names gives back a list with all the restaurants that have all those features. It also has a function that calculates the “closest” restaurant, which takes a list of feature names and finds the restaurant that has the most of them while having the least extra features (Using Closest Nearest Neighbor).
    - *Restaurant*: a class that contains the name, the city name, a vector of features and the IDs of the special features. The Closest Nearest Neighbor is done by passing a city a list of features and getting in return the distance.
    - *City*: they have been unused in the last build, but their purpose was just to have a vector of restaurants.

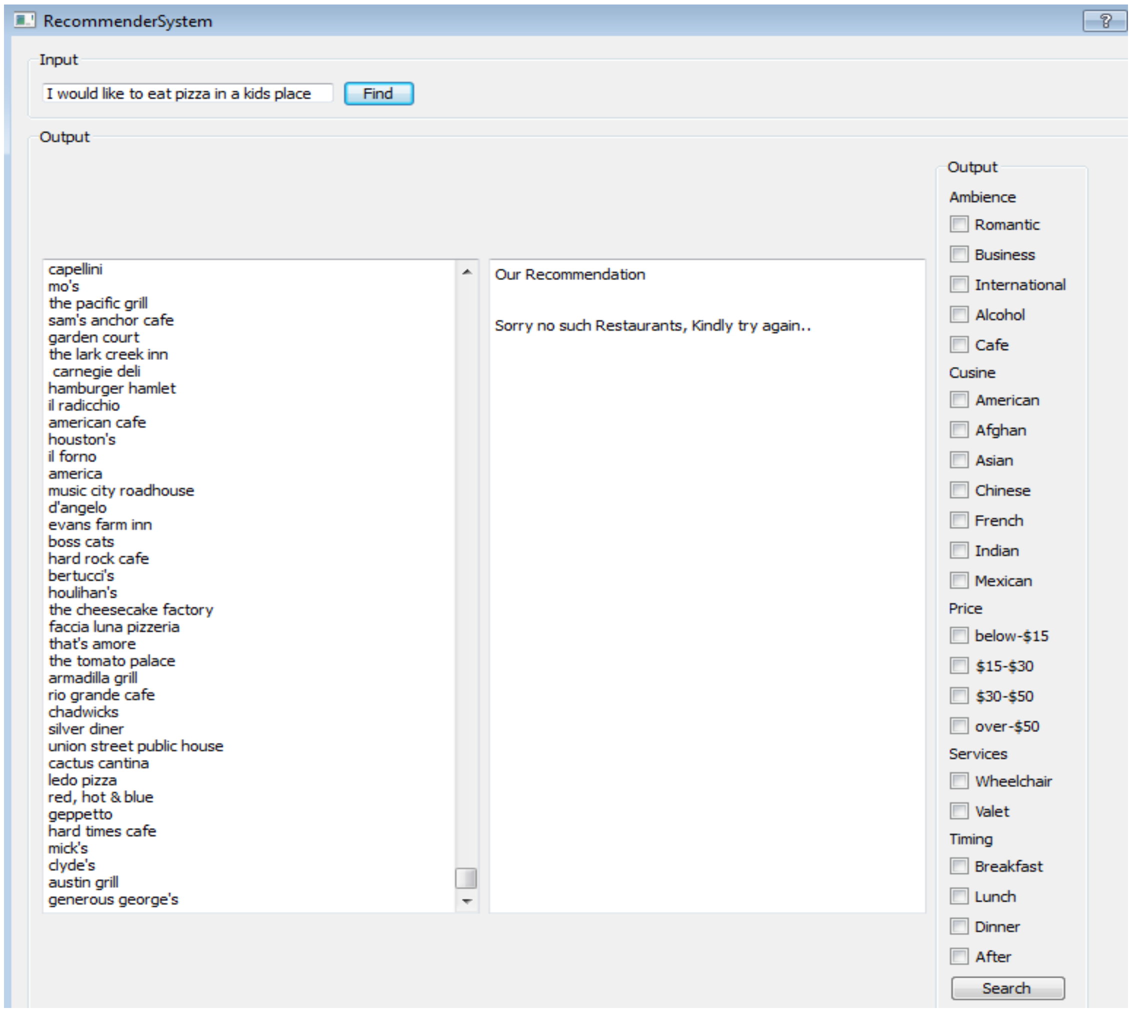
- **Closest Nearest Neighbor:** when no restaurant matches the desired features (no perfect match), this algorithm is used to find the restaurant that better fits the requirements. It basically increases the distance for each missing feature, and also adds a little distance for every extra feature (because they might not be desired features). Gives back the best three fits.

- **Searching:** Customer can enter search string to search for restaurants. Based on the keywords from the search string, search query is prepared. This search query may or may not match the feature list of data base. An approximation is performed to find the feature closest to the feature.

- **How to use:** Simply type in a sentence about what kind of restaurant, food or service you are looking for.



You will get a list of restaurants that have some or all of the identified features. Some checkboxes will prompt to help you further refine your search: select them and run the system again to get a better approximation.



I would like to eat pizza in a kids place

capellini
mo's
the pacific grill
sam's anchor cafe
garden court
the lark creek inn
carnegie deli
hamburger hamlet
il radicchio
american cafe
houston's
il forno
america
music city roadhouse
d'angelo
evans farm inn
boss cats
hard rock cafe
bertucci's
houlihan's
the cheesecake factory
faccia luna pizzeria
that's amore
the tomato palace
armadilla grill
rio grande cafe
chadwicks
silver diner
union street public house
cactus cantina
ledo pizza
red, hot & blue
geppetto
hard times cafe
mick's
clyde's
austin grill
generous george's

Our Recommendation

Sorry no such Restaurants, Kindly try again..