**We Complete Each Other's Sandwiches**

**Final Project: a Needle In a Data Haystack, HUJI**

[Daniel Gissin](mailto:daniel.gissin@mail.huji.ac.il), [Keren Luzon](mailto:keren.luzon@mail.huji.ac.il), [Clara Herscu](mailto:clara.herscu@mail.huji.ac.il)

**Problem description:**

Given a user's ingredients list our job was to supply new interesting ingredients that work well with his existing list, along with recipe suggestions.

**Data:**

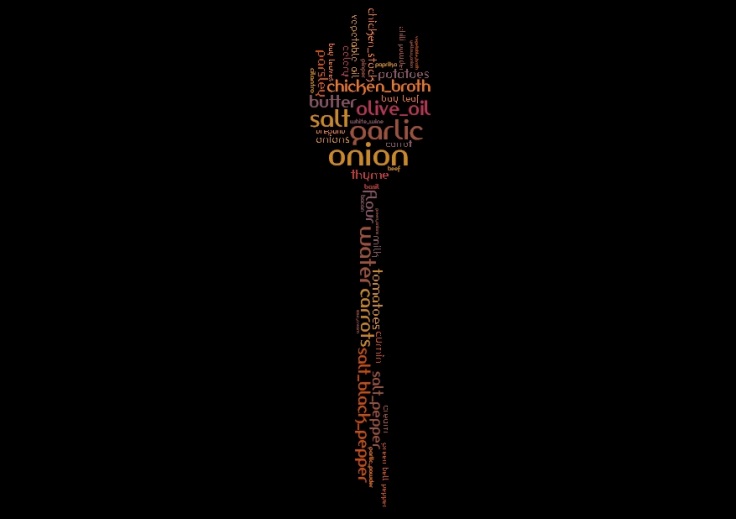
Our data consists of ~100,000 recipes, crawled from two websites[[1]](#footnote-1) (~100MB). For each one of the recipes we obtained the ingredients list, along with the rating and the number of reviewers if available.

*Pre-processing:*

While one of the website's data was cleaner, the other one needed quite a lot of pre-processing: many ingredients were described by full, long, sentences which needed to be reduced to a few words (i.e. "2/3 cup unsalted butter, cut in small chunks, plus, more for the pan" "unsalted butter"). Keren, how did we do that?

*Sanity checks and visualizations:*

To visualize which ingredients where dominant for each recipe category, we created word clouds as seen in figures 1-3.

Figures 1+2: Drinks and Soup Ingredients Word Cloud



Figure 3: All Categories Ingredients Word Cloud

**Solution:**

*General description:*

We modeled our problem as an association-rule mining problem, where association rules are from a set of ingredients to a single ingredient (the one we suggest adding). In order to mine the association rules, we used the A-Priori algorithm to find the support of all the relevant subsets of ingredients, and for the subsets that had a support higher than our defined threshold, we extracted the rules that had sufficient confidence. Our data differs from the model taught in class mainly in the fact that the baskets (single recipe ingredients list) are weighted (using rating and number of reviewers).

*Defining the weights:*

In approaching defining the weights of our baskets (recipes), we had to take a few things under consideration:

First, preserving monotonicity (downward closure property): adding an ingredient to a set of ingredients, cannot result in a higher support. This is necessary in order for the A-Priori algorithm to find all relevant frequent itemsets. This means we couldn't use negative weights (for recipes that are poorly rated), as this would allow a situation where adding an ingredient to a set removes negatively weighted baskets from the set's support, resulting in an increase in the overall support. The decision not to include negative weights also makes sense in our specific context, since we cannot infer causality between a combination of ingredients in a recipe and the recipe being poorly rated.

Second, while trying to encapsulate both rating score and number of reviewers, we wanted to create better separation between different rating scores in order to avoid a situation where a poorly rated recipe with many reviewers would weight more than a positively rated recipe with fewer reviewers.

Show distribution of rate scores

In order to achieve that better spread of ratings we defined: .

Finally, we defined the basket weight as:

We chose after trying several options and seeing that this one produced the best results for us, and we used the since the number of reviewers ranged from 0 to 1,000.

*Defining the support, confidence and interest:*

In order to have the support reflect the weights while still having it range from 0 to 1, we defined the support as follows:

The confidence was defined as defined in class, given this new definition of support:

Since we had a very large amount of distinct items (ingredients) in our dataset, the support of a single item was usually very low in comparison to the average confidence of an association rule. This brought us to change the definition of interest in order to make sure we aren't offering the user obvious ingredients (like sugar, salt & pepper, etc.). After trying a few different definitions of interests, the following one seemed to produce the best results:

*Tying it all together:*

We generated all the association rules that had a high enough confidence XXX, calculated their interests, and saved all the results to later serve as a static DB. Then, given ingredients from the user, we offered ingredients that both had high interest, and were not originally given by the user. The score according to which the ingredients where shown was determined to be:

*UI:*

Our user interface consists of a website where the user can write his ingredients and ask us to spice up his recipe. The user can also focus the search on recipes of a certain category or, if he's feeling spicy, let us find interesting ingredients from all categories.

*Start the fun here!*

**Experiments**

**Future Work:**

Since our project was based on only two websites which were both American, our data was biased towards American cuisine. An interesting direction to go with this project is to look at other cuisines.

**Brief conclusion**

1. <http://allrecipes.com/>; <http://www.foodnetwork.com/>; [↑](#footnote-ref-1)