ISYE 6740 Summer 2023 Project Proposal: “Augmented recipe recommendation using flavor profile, ingredients, and cooking technique”

Micaela McCall

July 03 2023

1 Background and Literature Review

As the amount of information on the internet has ballooned, recommendation systems have become increasingly crucial to help users find desired information without having to do extensive manual search. The goal of a recommender system is to predict the rating a user would give to a new item and to suggest to the user items for which the predicted rating is high. Food and recipe recommendation is a domain in which these systems are particularly relevant given the vast number of online recipes. Recipe recommendation has gained traction in the healthy-eating community as a way to suggest healthier meals or ingredient substitutions [1, 2, 3]. It is also a relevant context for group recommender system research; i.e., how to suggest recipes based on the preferences of all members of a group [4].

In addition to its many applications, basic recipe recommendation offers fertile ground for researching the nuances of recommendation systems because of the rich information about recipe contents (ingredients), types of recipes (breakfast, dinner, desert, etc.), styles of cooking, nutritional contents, etc. Some relevant questions in this area are: How do we quantify a user’s preference for specific ingredients given their rating of a recipe and leverage this information to improve recommendation? How do we incorporate preferences for particular types of recipes? How do we build recommender systems under constraint, such as only recommending recipes that conform to dietary restrictions?

Researchers have tried to address some of these questions through iteration on various approaches to algorithmic recommendation. Some of the most common are:

Collaborative filtering (CF) is a method that uses the ratings of many users over many items to identify similar users and predict the rating a user would give to an item based on the ratings given by similar users. The only data necessary for this approach is ratings history on items [5].

Content-based (CB) is a method that uses information about items to calculate similarity between new items and items a user has historically rated to predict the rating a user would give to an item [1]. For example, in the recipe/food domain, Freyne and Berkovsky (2010) broke down ratings for a recipe into ratings for ingredients and then reconstructed a prediction for a new recipe using a user’s ratings for the constituent ingredients.

Knowledge-based (KB) is a method that uses content knowledge about the item as well as knowledge about users needs or a set of constraints to recommend specific items and then includes an iterative process of eliciting users’ feedback. [1, 7]

Hybrid approaches are particularly used in recipe recommendation because we have both plentiful user ratings data and about item content. These approaches aim to combine the strengths of multiple previously mentioned approaches. [1, 7]

In my literature review, I noticed that many attempts to recommend healthy foods used a KB approach;

they didn't just reflect the user's preferences, but also the nutritional needs of the particular user based on their demographic information, e.g. in Alberg (2006). Attempts to conform to dietary restrictions were treated similarly [8]. However, I didn't find instances where rich recipe-based metadata was included in recommender algorithms to try to improve the quality of the recommendation, other than in Freyne and Berkovsky (2010).

2 Problem Statement

The aim of this project is to explore and compare the performance of CB, CF, and hybrid recommender algorithms in the context of recipe recommendation, while leveraging recipe flavor profiles and recipe metadata (meal type, cooking technique, cooking time) in the recommendation.

In contrast with some of the approaches mentioned above, I didn't search for access to a data set where recipe ratings for individuals with rich person-centered metadata was available. Rather, my goal is to use solely recipe-related metadata, and enhance this with a second data source of flavor profiles, directly in the implementations of CB, CF, and hybrid recommender algorithms.

Additionally, in the context of CB recommendations, I want to expand on the approach used by Freyne and Berkovsky (2010) to include recipe flavor and metadata.

3 Data Sources

The first data set is a set of recipes from Food.com, made available on as a Kaggle dataset (`https://www.kaggle.com/datasets/shuyangli94/food-com-recipes-and-user-interactions?resource=download&select=RAW_recipes.csv`). The recipes data ncludes the name of the recipe, a description of the recipe, recipe tags, the nutritional value, the steps in make the recipe, and the ingredients.

The second data set is a set of recipe interactions also from Food.com and also made available through Kaggle (`https://www.kaggle.com/datasets/shuyangli94/food-com-recipes-and-user-interactions?resource=download&select=RAW_interactions.csv`). This data set includes the rating and review given to a recipe by various users (recipe id matches up to the recipe id in the recipes data set above).

The third data set provides the flavor molecules and associated flavor profile for a given food (from `https://cosylab.iiitd.edu.in/flavordb/`). It is accessed via API calls for each ingredient in the recipes from the first datset.

A final data set provides ingredient lemmatization, or in orther words associating each possible ingredient with a "base" version of that ingredient (e.g. all types of lettuce become "lettuce"). Also made available through Kaggle (`https://www.kaggle.com/datasets/shuyangli94/food-com-recipes-and-user-interactions/discussion/118716?resource=download&select=ingr_map.pkl`).

4 Methodology

4.1 Data Preprocessing

For the recipe data set:

- Ingredients will be extracted and associated with their "base" ingredient per the ingredient lemmatization map.
- Tokenization of recipe tags.
- Cooking techniques extracted from recipe steps.

For data fusion:

- Each ingredient in each recipe queried for flavor profile.
- Flavor profile of each ingredient aggregated across recipe.
- Flavor profile associated with respective recipe.

Train-test split:

- For interactions data set, data will be split into training, evaluation, and testing data sets. The interactions data will be the source of ground-truth that I will use to evaluate the recommendations. All/any recipes could be present in each of the train, eval, and test interactions data sets.

4.2 Algorithms

Baseline model

First, I plan to build a model that randomly predicts ratings for a given user on a given recipe in order to compare the other algorithms used to this baseline.

Collaborative filtering (CF)

Calculate the nearest neighbors of a new user measured by cosine similarity:

$$Sim(u_i, u_k) := \frac{r_i * r_k}{|r_i| * |r_k|} = \frac{\sum_{j=1}^{m} r_{ij} r_{kj}}{\sqrt{\sum_{j=1}^{m} r_{ij}^2 \sum_{j=1}^{m} r_{kj}^2}} \tag{1}$$

where r_i and r_k are ratings vectors for users u_i and u_k.

Predict a user's rating on a new recipe r_{ij} by weighted average with bias avoided by by subtracting each user's average rating $\tilde{r_k}$ from their rating of the recipe and adding in the target user's average rating $\tilde{r_i}$:

$$r_{ij} = \tilde{r_i} + \frac{\sum_k Sim(u_i, u_k)(r_{kj} - \tilde{r_k})}{\text{num ratings}} \tag{2}$$

Content-based (CB)

A rating for each user on each ingredient is calculated as the average of the ratings each user gave to all recipes including that ingredient:

$$rat(u_i, ingr_j) = \frac{\sum_{l;ingr_j \in l} r_{il}}{l} \tag{3}$$

where r_{il} is the rating user i gave to recipe l. This formula is then applied over the flavor profile of each recipe, tags, and cooking techniques, to create a comprehensive recipe-based data source for each user. Predict a user's rating on a new recipe r_{ij} by finding the average rating across all the ingredients, flavors, and cooking techniques in the new recipe:

$$r_{ij} = \frac{\sum_{l \in rec_j} rat(u_i, ingr_l)}{l} \tag{4}$$

Hybrid

I plan to try two approaches here.

Content-augmented CF using cosine similarity: will be the same as the CF approach except instead of using ratings on recipes in Equation 1, I will use ratings on ingredients, flavors, and techniques.

Content-augmented matrix factorization: takes the matrix factorization approach to CF and augments the item data with content, as described below. Matrix factorization aims to decompose the user's preferences for into preferences for a set of latent factors. Matrix factorization can be performed using Singular Value Decomposition (SVD):

$$M = U\Sigma V^T \tag{5}$$

By selecting the top k singular values of matrix Σ, we can reconstruct matrix M with less dimensions but still capturing much of the variability of the original matrix [9]. The idea here, when applied over flavor ratings, would be to find the dimensions of latent flavor preferences so as to avoid having to deal with the high dimensionality of flavor profiles. However, when factoring a sparse matrix, it's more efficient to use Non-negative Matrix Factorization (NMF), which involves solving the following optimization problem to find U and V such that the reconstructed user-item rating $\hat{r}_{ij} = u_i^T v_j$ is as close as possible to the true r_{ij}:

$$min_{q,p} \sum_{(i,j)\in TR} (r_{ij} - u_i^T v_j)^2 + \lambda(|u_i|^2 + |v_j|^2) \tag{6}$$

where u_i is the user vector, the i-th row of matrix U, and v_j is the item vector, the j-th row of matrix V, and TR is the training set [9].

To reconstruct a rating prediction using this method and incorporating recipe content information (i.e. flavor profile), the item vector becomes:

$$V = X\Phi \tag{7}$$

My source for this approach is Forbes et al. (2011). This changes Equation 6 slightly

$$min_{q,p} \sum_{(i,j)\in TR} (r_{ij} - u_i^T \Phi^T v_j)^2 + \lambda(|u_i|^2 + |\Phi|^2) \tag{8}$$

To solve this NMF, Alternative Least Squares is typically used [9]. The gradient with respect to u_i and the gradient with respect to Φ are computed and alternately applied to minimize the objective.

5 Evaluation and Final Results

I plan to use the Root Mean Square Error (RMSE), Mean Absolute Error (MAE) and coverage (ability to generate predictions) [6] to evaluate the performance of the algorithms. These metrics will be applied by comparing the true ratings from the evaluation set to the predicted ones. For my final result, I will report the metrics for the baseline model, the vanilla CF model, the CB model, the hybrid CB-CF model, and the hybrid matrix factorization model. My aim is to analyze the performance of these approaches to evaluate whether the inclusion of flavor profile, ingredients, and cooking technique improve the quality of the recommender system.

6 References

1. Trang Tran, T. N., Atas, M., Felfernig, A., Stettinger, M. (2018). An overview of recommender systems in the healthy food domain. *Journal of Intelligent Information Systems*, 50, 501-526.

2. Pecune, F., Callebert, L., Marsella, S. (2020, September). A Recommender System for Healthy and Personalized Recipes Recommendations. In *HealthRecSys@ RecSys* (pp. 15-20).

3. Van Pinxteren, Y., Geleijnse, G., Kamsteeg, P. (2011, February). Deriving a recipe similarity measure for recommending healthful meals. In *Proceedings of the 16th international conference on Intelligent user interfaces* (pp. 105-114).

4. Masthoff, J. Group recommender systems: Combining individual models. *Recommender systems handbook*, Springer 677–702 (2011).

5. Ajitsaria, A. Build a Recommendation Enging with Collaborate Filtering. *RealPython.com*

6. Freyne, J., Berkovsky, S. (2010, February). Intelligent food planning: personalized recipe recommendation. In *Proceedings of the 15th international conference on Intelligent user interfaces* (pp. 321-324).

7. Burke, R. Hybrid Recommender Systems: Survey and Experiments. *User Model User-Adap Inter* 12, 331–370 (2002).

8. Aberg, J. (2006, January). Dealing with Malnutrition: A Meal Planning System for Elderly. In *AAAI spring symposium: argumentation for consumers of healthcare* (pp. 1-7).

9. Luo, S. (2018, December). Introduction to Recommender System Approaches of Collaborative Filtering: Nearest Neighborhood and Matrix Factorization. *towardsdatascience.com*

10. Forbes, P., Zhu, M. (2011, October). Content-boosted matrix factorization for recommender systems: experiments with recipe recommendation. In *Proceedings of the fifth ACM conference on Recommender systems* (pp. 261-264).