CS 235: Midterm Report

Health Based Ingredient Recommender System for Recipes

1. Introduction

1.1. Project Type: Research 1

1.2. Problem Statement:

Food, healthy eating and new recipes have become central subjects in our daily life. Specially with obesity being a major health problem across the world, specially in the United States and an increased public interest in healthy lifestyles, the study of techniques to assist people following a healthy diet is significant. Most of the work that has been done in the domain of food and recipes are on suggesting ingredients to recipes or improving recipes. But one thing to keep in mind is that, adding any particular ingredient to a recipe will change the nutrition measurement of that recipe. So, it might happen that, suggested ingredient(s) may not be healthy for people. The works done so far does not take into consideration about the health and nutrition facts.

1.3. Project Aim:

We aim to build a recommender system that will solve the problem at hand by completing partial recipes based on health component values of ingredients. The system will calculate individual health value (calories, protein, fat, sodium counts) and check whether the recommended ingredients increase the health value of recipes. If yes, then it will suggest.

2. Update on Mid-term Report:

Initially we decided to work with fusing two datasets because we both needed the recipe title including ingredients along with their health values. After doing some initial experiments, we found out that one of the dataset is difficult to work with. So we have decided to work with only the “epicurious” dataset [12] for our system.

3. Related Work Survey

3.1. Survey by Muhammad Shihab Rashid:

In [1], the authors proposed a recommender system to suggest ingredients that can be added to a partial recipe based on users’ ratings of a particular recipe. For that, they used item-based collaborative filtering algorithm, using a high-dimensional, sparse dataset of recipes, which inherently contains only implicit feedback. CF uses a matrix of all items and users to produce recommendations, where typically each element of the matrix represents how much a user likes an item. They implemented item-based CF by first calculating the similarities between all ingredients. Then they computed the fit of an ingredient to a recipe by summing the similarities of the target ingredients’ closest neighbors that are present in the target recipe and then scaling by the sum of all similarities in the target ingredients’ neighborhood. They also used PCA to reduce the number of dimensionality to speed up the computation. Their results conclude that using the PCA reduced data consistently outperformed the non-reduced data. The **advantages** of item-based CF methods can be scaled to large datasets and are well suited for sparse data. They also do not rely on item content, they are easy to implement and new data can be added incrementally. The **disadvantages** are that the recipes do not contain explicit feedback, the ratings are only binary. Negative rating of any recipe is not present in the dataset. The possible **extension** of this dataset is to include negative reviews of recipes and integrate with CF methods to generate even better ingredients.

In [2], the authors proposed a recommender system that not only offers recipe recommendations that suits the user’s preference but is also able to take the user’s health into account. They have designed a complete interaction process that includes preference elicitation, recommendation generation and presentation, user support for browsing and critiquing the given recommendations as well as for providing the user alternative recommendations. Based on the user profile, the long term preference and preference elicitation, the system generates a set of recipe generations and present them one by one, which later the user rates and tags. The

recommendation is done by extending matrix factorization by including additional parameters to model dependencies between assigned tags and ratings. They have created a health component, which is based on a calorie balance function of the difference between the calories the user still needs and the calories of that recipe. The **advantage** of this paper is that, it is one of the first papers to recommend keeping the balance between taste (preference) and health factor. Their entire platform is based on user/recommender interaction design from preference elicitation which is also novel. The **disadvantage** is that their health component only takes into factor how much calorie the user needs rather than computing the individual calories of the ingredients. Possible **extension** of this paper can be to calculate the individual health values of ingredients and take that into factor while recommending to the user.

3.2. Survey by Quazi Mishkatul Alam:

In [3], the authors suggest numerous works on the problem of identifying user patterns and consumption preferences, and using collaborative filtering or content-based filtering to suggest ingredients in recipes. This paper tackles the unique problem of recommending missing ingredients to a recipe or creating a recipe of its own (in contrast with the user centric approach). Two deep learning methods are used for this purpose, namely, Non-negative matrix factorization and two-step regularized least squares. The first method is capable of using information in the existing recipes (ingredient combinations that often occur together) to complete a recipe. However, it cannot find recipes that are not present in the dataset. This limitation is overcome with the second method, which will not only suggest missing ingredients to a recipe, but also suggest new ingredient combinations to create a new recipe. The data used in this paper is taken from Ahn et al. (2011), which apart from having the recipes and ingredients information, also has flavor components information. The (recipe, ingredients) data is preprocessed into a m x n matrix where m is the number of recipes and n is the number of ingredients. The (ingredient, flavor component) data is preprocessed into a p x m matrix where p is the flavor components. After, that the two methods are applied using these two matrices. This paper manifests that deep learning algorithms can be used to complete recipes. The first method, namely, Non-negative matrix factorization is particularly perfect for this type of data. This is due to the fact that the data is a linear combination of non-negative (0/1) values which depicts whether or not an ingredient is present in the recipe. However, this method has two disadvantages. Firstly, it cannot take into account the flavor components information which can be quite important in food related predictions. Secondly, it cannot suggest a new recipe which is not present in the dataset. The second method, namely, two step regularized matrix factorization takes care of these limitations. **Pros:** i) This paper gives us two methods that can be used to implement the solution to our problem (completing a recipe). **Cons:** i) One caveat is that it doesn't suggest the ingredients that uplifts the health values, rather the association among the ingredients present in the existing recipes is considered only.

This paper [4] emphasizes on the fact that certain factors affect the ingredients that can be used to together in a recipe, such as, color, temperature, texture, sound, etc. However, palatability largely depends on the flavor components. Therefore, it can be said that the flavor components are an incisive factor when analyzing our acceptable combination of ingredients in a recipe. A hypothesis that is the center point of this paper is, "ingredients sharing flavor compounds are more likely to taste well together than ingredients that do not". To realize this the researchers adopt a network based approach to analyze the effect of flavor components on ingredient combination. Keeping a set of ingredients in one column and a set of flavor component in another column a bipartite network is formed. A projection of this bipartite network is the so-called flavor network in which the ingredients are the nodes and two nodes have an edge if the share at least one flavor component. In fact, the weight of the edges is considered to be the number of shared

flavor component. In this paper these facts are brought to light that the number of ingredient and the rank-frequency behavior of ingredients in recipes among different cuisines are invariant. Also, it also discusses about how some ingredients are prevalent in a recipe, while the others are not so much. **Pros:** This paper brings to light a number of significant patterns that characterizes our preference of combining ingredients to form a recipe **Cons:** Doesn't shed any light on any methods that we could use, rather statistical analysis is presented.

3.3. Survey by Marc Giannuzzi:

The project represented in [5] has for purpose to recommend recipes for users. The datasets that have been used are Epicurious and Food.com. Many data mining techniques are utilized such as SVM (Support Vector Machines) and TD-IDF weight. What is interesting in their approach is the adaptation of the Rocchio's algorithm for food recommendation. This algorithm enables to have some feedback on the documents used and on how to give documents that satisfy most the user. Thus, the Rocchio's algorithm and the term frequency can be used to recommend not documents but recipes that fit with the user desired ingredients which are used as the terms or words. In fact, a recipe whose weight would not be recommended whereas a positive one would be. Also, the more the absolute value of the weight is near 1, the more the result is accurate and can be used. According to the writer of the paper, Jorge Almeida, that way of recommending brought good results. However, he wanted to know how to make results even better. What he did is to add two methods which are Min-Max Normalization in order to translate the similarity that just had been computed into ratings and another one that uses deviation. To be even more precise, a cross validation has been done with the several methods used.

In [6], Four students worked on how to recommend food based on rates of recipes, the user age, gender and other characteristics in the purpose to answer the question: "What should I prefer to eat" and then recommend a restaurant to the client. They used neural networks to implement their recommendation system. Neural networks are based on two different steps: feed forward algorithm and back-propagation algorithm that are working on several layers that include neurons. Connections are made between each neuron of two layers that are closed with weights. The main purpose is to calculate errors with different entries to obtain a model that can fit with the one that the students have chosen. Each time an error is computed for a specific entry, the weights that are described before are updated in order to achieve the model that is wanted. Thus, the group of students developed an Android application which purpose is to recommend what food to eat to the clients. Another application has been made especially for the owner of the hotel who can feed the database with it's own recipes and ingredients. Also, an admin application has been made in order to add or retrieve some restaurants from the client application.

3.4. Survey by Quentin Lacroix:

The paper [7] states: (1) an analysis reporting on the applicability of various personalized techniques for rating prediction, and (2) a report on the observed trends of reasoning uncovered by machine learning feature selection algorithm. Related work: Asked recipe ratings, which are transferred to ingredient ratings, is an accurate and effective method of capturing ingredient preferences -> accuracy in recommendations. Extracting important features from the recipe text -> weighted similarity measure between recipes. 3 personalized recommender algorithms: 2 recommender strategies: collaborative filtering algorithm assigns predictions for user for a target recipe, based on the weighted ratings of a set of neighbors and content-based algorithm : breaks down each rated (1 user rate) recipe into rated ingredients, making the target recipe. There is also a machine learning strategy for rating prediction: logistical binary decision tree to predict scores based on the recipe content and metadata + splits after testing all features and computing each expected reduction in error + each leaf predicts a numeric quantity using linear regression. Algo accuracy and performance is assessed with recipe ratings from Mechanical Turk (amazon)

(101,557 ratings of 917 users). Mean Absolute Error smaller in ML algo. **Pros:** In collaborative filtering, all the stakes and potential errors come from the number of ratings of the user (or users), therefore we can get good accuracy on recommender algorithms after a while and everybody eats 3 times a day: we can be optimistic. **Cons:** many factors that have an impact on a user's opinion on foods (like trends etc) are tough to associate. "Recipe diversity could depend on the user, rather than just on the recipe similarity". **Possible extensions** may be others analysis providing an artificial reasoning, approaching human reasoning.

In [8] , apart from **content-based** (*features of the food/item to generate user profiles and then check for similarities*), **collaborative filtering** (*other people having similar tastes*), **rule-based approach** (*rules derived from previous transactions and associations*), and **hybrid** (*multiple techniques using time, similar taste, associations rules*), the authors focus on a different type of recommendation system: Sequential Pattern Mining. The purchase history of users is analyzed to find their sequential patterns (**SPADE** algoritm i.e Sequential PAttern Discovery using Equivalence classes) to predict the next possible food purchase. Suggest food based on previous transactions can help customer gain time. We can do better Ads and better management of food storing. Transaction history of a user is a series of transactions ordered by the transaction time. 3 Steps to do with this transaction history: collect user infos and preferences, then find sequential patterns from previous transaction stored in the database. After that one of these patterns will be shown as a special offer with discount: if the customer purchase it, the preference is stored; if not ask for feedback. Results of the experiments is well known now but: when minimum support is increased, number of patterns generated decreased. We can predict food-next purchase with sequential patterns. **Pros**: It can be useful in the same location. Indeed, at the supermarket near people's home, they should buy pretty much the "same" daily things, so we can trust frequent itemsets. **Cons**: Useful feedback is difficult to obtain. Sequential patterns may not always be interesting because of a lack of correlation between items, just bought together. We have to **extend** bought ingredients patterns to recipes and it depends on the person.

3.5. Survey by Kristrian Tram:

The authors in [9] focus on the idea of trying to detect the ethnic cuisine type of food based on the similar ingredients. The authors found that using an SVD worked well to reduce the dimensionality of each recipe, finding the sweet spot of 400 kinds of ingredients. They used each ingredient as a separate feature, but struggled to classify recipes with many simple and common ingredients. They then looked to use an SVM method which proved to do very well in classifying the various cuisine types. Combining both methods they could increase the precision of classifying the recipes by cuisine. The authors focused on the precision and recall curves to evaluate the accuracy of their classifier. Using 5-Fold cross validation the authors found their classifier to have an 80%+ precision score for all of the cuisines they classified. The authors felt that with the increasing number rid recipe sharing online, a useful feature would to be able to automatically group recipes that are similar cuisines to better recommend recipes to others. They believe this to be very beneficial in websites that focus highly on recommended similar recipes.

The authors in [10] spent much of their efforts gathering and cleaning data to get a representative dataset that's was clearly classified. The authors used data from recipesource.com and various other American recipe websites. The authors had to reduce their dataset of over 70,000 recipes to a mere 5,900 recipes as the source they used allows anonymous submissions without strict guidelines on classify recipes. With their limited data the authors gathered only recipes from countries that had at least 50 recipes. One major issue that the authors faced was skewed data. As certain Ingredients such as Soy sauce is very prominent in Asian countries with as much as a 30% more prevalence then European countries. In contrast many Europeans countries shared many of the same ingredients. The authors used Hierarchical clustering based on ingredient

similarity. The authors chose the Euclidean distance based on the prevalence of ingredients. Using the similarities of ingredients and super catagories of the countries, the authors created ingredient networks to identify groups of ingredients to predict a country of origin. The authors train their classifier by splitting the data into 5 parts and training on 4 parts with a single test set. They found their classifier to be accurate in the 80 and 85 percentile in Asian and European ingredients.

4. Project Progress Report

4.1. Work by Muhammad Shihab Rashid:

Worked on dataset cleaning and preprocessing. We require such datasets that would contain recipe titles, their ingredients alongside their health values. But the [12] dataset contains a lot of noise as seen in Figure 1. It contains redundant feature columns such as tags. The redundant features were removed from csv so that only the ingredients remained. Also, the rows with missing data were removed. Generated a new cleaned csv file.

	title	rating	calories	protein	fat	sodium	#cakewee	#wasteles	22-minute	3-ingredie	30 days of advance p	alabama	alaska	
2	Lentil, Apple, and Turkey W	2.5	426	30	7	559	0	0	0	0	0	0	0	0
3	Boudin Blanc Terrine with R	4.375	403	18	23	1439	0	0	0	0	0	0	0	0
4	Potato and Fennel Soup Ho	3.75	165	6	7	165	0	0	0	0	0	0	0	0
5	Mahi-Mahi in Tomato Olive	5					0	0	0	0	0	0	0	0
6	Spinach Noodle Casserole	3.125	547	20	32	452	0	0	0	0	0	0	0	0
7	The Best Blts	4.375	948	19	79	1042	0	0	0	0	0	0	0	0
8	Ham and Spring Vegetable S	4.375					0	0	0	0	0	0	0	0
9	Spicy-Sweet Kumquats	3.75					0	0	0	0	0	0	0	0
10	Korean Marinated Beef	4.375	170	7	10	1272	0	0	0	0	0	0	0	0
11	Ham Persillade with Mustar	3.75	602	23	41	1696	0	0	0	0	0	0	0	0
12	Yams Braised with Cream, R	3.75	256	4	5	30	0	0	0	0	0	0	0	0
13	Spicy Noodle Soup	4.375					0	0	0	0	0	0	0	0
14	Banana-Chocolate Chip Cak	4.375	766	12	48	439	0	0	0	0	0	0	0	0
15	Beef Tenderloin with Garlic	4.375	174	11	12	176	0	0	0	0	0	0	0	0

Figure 1: Noisy dataset with redundant features

	title	rating	calories	protein	fat	sodium	almond	anchovy	apple	apricot	artichoke	arugula	asian pear	asparagus	avocado
2	Lentil, Apple, and Tur	2.5	426	30	7	559	0	0	1	0	0	0	0	0	0
3	Boudin Blanc Terrine	4.375	403	18	23	1439	0	0	0	0	0	0	0	0	0
4	Potato and Fennel So	3.75	165	6	7	165	0	0	0	0	0	0	0	0	0
5	Spinach Noodle Casse	3.125	547	20	32	452	0	0	0	0	0	0	0	0	0
6	The Best Blts	4.375	948	19	79	1042	0	0	0	0	0	0	0	0	0
7	Korean Marinated Be	4.375	170	7	10	1272	0	0	0	0	0	0	0	0	0
8	Ham Persillade with I	3.75	602	23	41	1696	0	0	0	0	0	0	0	0	0
9	Yams Braised with Cr	3.75	256	4	5	30	0	0	0	0	0	0	0	0	0
10	Banana-Chocolate Ch	4.375	766	12	48	439	0	0	0	0	0	0	0	0	0
11	Beef Tenderloin with	4.375	174	11	12	176	0	0	0	0	0	0	0	0	0
12	Peach Mustard	3.125	134	4	3	1394	0	0	0	0	0	0	0	0	0
13	Raw Cream of Spinacl	4.375	382	5	31	977	0	0	0	0	0	0	0	0	0

Figure 2: Cleaned Dataset

In Figure 2, we can see that only the columns with ingredient names remains and there are no rows with missing values.

4.2. Work by Quazi Mishkatul Alam:

Worked on our initial dataset [11] that only contains ingredients but not the health values. As it is huge, used a subset of data (1000 recipes/100 ingredients). Augmented the dataset by leaving out some of the ingredients of the original recipe, but using the full recipe as the label, both represented by a 0/1 vector. Used a basic neural network model in Keras to train model that suggested top 8 values in output vector for suggestion.

4.3. Work by Marc Giannuzzi:

Worked on analyzing the “simplified dataset” [11] to view the recipes inside it and work on it later. “npz” files cannot be opened directly so it had to be visualized.

```
C:\Users\Marc\Desktop\ESILV\UCR\Serious\Projets\CS235>py script_visu_simplified_recipes.py
Recipe number 1 :
        Ingredients:

['basil leaves' 'focaccia' 'leaves' 'mozzarella' 'pesto' 'plum tomatoes'
 'rosemary' 'sandwiches' 'sliced' 'tomatoes']


Recipe number 2 :
        Ingredients:

['balsamic vinegar' 'boiling water' 'butter' 'cooking spray'
 'crumbled gorgonzola' 'currants' 'gorgonzola' 'grated orange' 'kosher'
 'kosher salt' 'orange rind' 'parsley' 'pine nuts' 'polenta' 'toasted'
 'vinegar' 'water']


Recipe number 3 :
        Ingredients:

['bottle' 'bouillon' 'carrots' 'celery' 'chicken bouillon' 'cilantro'
 'clam juice' 'cloves' 'fish' 'garlic' 'medium shrimp' 'olive' 'olive oil'
 'onion' 'pepper' 'pepper flakes' 'red pepper' 'red pepper flakes' 'salt'
 'sherry' 'shrimp' 'stewed tomatoes' 'tomatoes' 'water' 'white'
 'white wine']


Recipe number 4 :
        Ingredients:

['grand marnier' 'kahlua']


Recipe number 5 :
        Ingredients:

['black pepper' 'coarse sea salt' 'fresh lemon' 'fresh lemon juice'
 'ground' 'ground black pepper' 'lemon' 'lemon juice' 'lime' 'lime peel'
 'mayonaise' 'pepper' 'sea salt' 'shallots' 'sherry wine'
 'sherry wine vinegar' 'vinegar' 'wine vinegar']


Recipe number 6 :
        Ingredients:

['black pepper' 'blue cheese' 'buttermilk' 'cheese' 'chives'
 'cider vinegar' 'cracked black pepper' 'pepper' 'ricotta'
 'ricotta cheese' 'roasted' 'roasted garlic' 'salt' 'sauce' 'vinegar'
 'worcestershire sauce']


Recipe number 7 :
        Ingredients:

['almonds' 'basmati' 'basmati rice' 'bay leaves' 'black cardamom pods'
 'brown' 'cardamom' 'cardamom pods' 'cashews' 'chili powder' 'chilies'
 'cilantro' 'cinnamon' 'cinnamon sticks' 'cloves' 'coriander'
 'coriander powder' 'cumin powder' 'ghee' 'green' 'green chilies'
 'hot water' 'leaves' 'milk' 'mint' 'mutton' 'oil' 'onions' 'paste'
 'powder' 'raisins' 'saffron' 'salt' 'tomatoes' 'turmeric'
 'turmeric powder' 'water' 'yogurt']
```

Figure 3 : Visualization of [11] dataset in console

Also worked on epicurious [12] dataset to visualize.

```
C:\Users\Marc\Desktop\ESILV\UCR\Serious\Projets\CS235>py script_visu_recipes_with_ratings_and_nutrition.py
Title : "Lentil
        Rating :  Apple
        Calories :  and Turkey Wrap "
        Proteins : 2.5
        Fat : 426.0
        Alcoholic : 0.0

Title : Boudin Blanc Terrine with Red Onion Confit
        Rating : 4.375
        Calories : 403.0
        Proteins : 18.0
        Fat : 23.0
        Alcoholic : 0.0

Title : Potato and Fennel Soup Hodge
        Rating : 3.75
        Calories : 165.0
        Proteins : 6.0
        Fat : 7.0
        Alcoholic : 0.0

Title : Mahi-Mahi in Tomato Olive Sauce
        Rating : 5.0
        Calories :
        Proteins :
        Fat :
        Alcoholic : 0.0

Title : Spinach Noodle Casserole
        Rating : 3.125
        Calories : 547.0
        Proteins : 20.0
        Fat : 32.0
        Alcoholic : 0.0

Title : The Best Blts
        Rating : 4.375
        Calories : 948.0
        Proteins : 19.0
        Fat : 79.0
        Alcoholic : 0.0

Title : Ham and Spring Vegetable Salad with Shallot Vinaigrette
        Rating : 4.375
        Calories :
        Proteins :
        Fat :
        Alcoholic : 0.0

Title : Spicy-Sweet Kumquats
        Rating : 3.75
        Calories :
        Proteins :
        Fat :
        Alcoholic : 0.0

Title : Korean Marinated Beef
        Rating : 4.375
        Calories : 170.0
        Proteins : 7.0
        Fat : 10.0
```

Figure 4: Visualization of Epicurious dataset in console

References:

[1] Cueto, P. F., Roet, M., & Słowik, A. (2019). Completing partial recipes using item-based collaborative filtering to recommend ingredients. *arXiv preprint arXiv:1907.12380*.

[2] Ge, M., Ricci, F., & Massimo, D. (2015, September). Health-aware food recommender system. In *Proceedings of the 9th ACM Conference on Recommender Systems* (pp. 333-334). ACM.

[3] De Clercq, M., Stock, M., De Baets, B., & Waegeman, W. (2016). Data-driven recipe completion using machine learning methods. *Trends in Food Science & Technology*, *49*, 1-13.

[4] Ahn, Y. Y., Ahnert, S. E., Bagrow, J. P., & Barabási, A. L. (2011). Flavor network and the principles of food pairing. *Scientific reports*, *1*, 196.

[5] Almeida, J. Personalized Food Recommendations.

[6] Patil, M. P. D., Patil, M. N. P., Geete, M. P. Y., Nikat, M. A. V., & Kulkarni, M. P. S. Food Recommendation System.

[7] Freyne, J., Berkovsky, S., & Smith, G. (2011, July). Recipe recommendation: accuracy and reasoning. In *International conference on user modeling, adaptation, and personalization* (pp. 99-110). Springer, Berlin, Heidelberg.

[8] Khandagale, S., Mallade, S., Kharat, K., & Bansode, V. (2016). Food recommendation system using sequential pattern mining. *Imperial J. Interdiscip. Res*, *2*(6), 912-915.

[9] Su, H., Lin, T. W., Li, C. T., Shan, M. K., & Chang, J. (2014, September). Automatic recipe cuisine classification by ingredients. In *Proceedings of the 2014 ACM international joint conference on pervasive and ubiquitous computing: adjunct publication* (pp. 565-570). ACM.

[10] Kim, K. J., & Chung, C. H. (2016). Tell me what you eat, and i will tell you where you come from: A data science approach for global recipe data on the web. *IEEE Access*, *4*, 8199-8211.

[11] https://dominikschmidt.xyz/simplified-recipes-1M/

[12] https://www.kaggle.com/hugodarwood/epirecipes