Steve Hall
James Crane
Bobby Albani
11 Dec 2024
2.155/156

Taste and Thrive: A Balanced Meal Plan

Abstract

The intent of this project is to develop a meal plan tool that incorporates a large corpus of externally sourced recipes and associated reviews to provide a diverse variety of quality meals that meet nutritional requirements set by the user, and that also taste good. The tool combines machine learning techniques with a constrained discrete optimization algorithm to generate a meal plan for the chosen period (e.g 1 day, 7 days, 14 days).

The inclusion of machine learning comes in the form of recipe analysis. A database of over 500,000 recipes with over 1.2 million reviews was analyzed to ensure quality (in the form of highly rated meals) and variety.

The methodology combined three key machine learning components: ratings normalization through weighting and sentiment analysis, text embeddings calculation for recipe similarity, and logistic regression for meal type classification. Optimization was conducted to maximize the normalized ratings average and minimize the similarity across the two-week plan while remaining within the nutritional constraints.

We demonstrate that the tool successfully generates a varied meal plan with high average meal rating while maintaining daily nutritional targets.

This project addresses what we believe to be an unexplored method of developing nutritious meal plans. The incorporation of constraint-based optimization on nutritional content may exist, but the extension to include taste and diversity based on more sophisticated recipe analysis and ensuring variety across meals provides a feature not apparent in other meal planning tools.

Introduction

Firstly, we would like to acknowledge how lucky we are to live in a time and country where food insecurity is low (<15%) with few citizens going truly hungry. However, despite the lack of hunger, one of the things that most Americans today struggle with is nutrition. Whether someone wants to lose weight, properly fuel their body for sports/physically demanding hobbies, or if someone simply wants to ensure they are getting the proper nutrients, finding the right way to achieve this can be difficult. It can be a challenge to ensure one consumes the correct number of “macros” (macronutrients, essentially the balance of carbohydrates, fat, and protein consumed) while remaining within the desired calorie range.

Food is more than just fuel; it plays a significant role in culture, social bonding, and personal satisfaction. Even more importantly, for nearly everyone, flavor matters (Mela, 1999). While there may be extremely dedicated people that see food as fuel, they are few and far between. The pleasure derived from eating a well-flavored, balanced meal can enhance overall well-being and make healthy eating habits more sustainable (Bublitz et al., 2010). Unfortunately, many of the foods in America today that have “the best flavor” are hardly the healthiest. Ultra-processed foods are linked with a myriad of chronic diseases, and many “tasty” treats are loaded with calories and have a terrible macro balance.

Typically, these challenges are addressed with solutions that are either blanket/non-targeted “fad” diets or through boring, bland, and repetitive meal preps of eating rice and chicken 4 days in a row. Ideally there could be a way for people to find a variety of nutritious meals that they enjoy, tailored to their nutritional requirements.

Health optimized meal planning ranges from manual planning where people count macros or calories to stay within a range each day to more advanced algorithms. Most of these advanced algorithms are some sort of constrained discrete optimization. For example, many algorithms optimize meal plans based on staying within a daily range of a combination of calories, ingredient types, or macros (carbs, fats, and proteins). Some even estimate how a food type will affect blood sugar level and optimize for a future range of blood sugar. There is one key problem with these techniques, people often eat food for enjoyment, not merely sustenance (Bublitz et al., 2010; Mela, 1999). While other algorithms may be able to account for taste by only including enjoyable recipes, they would have to not only carefully curate their recipe list but also may lack variety in their meal plans. We build on previous techniques by optimizing both health goals and taste, while also considering food diversity. This is particularly important, because adoption of a meal plan is often contingent upon enjoyment.

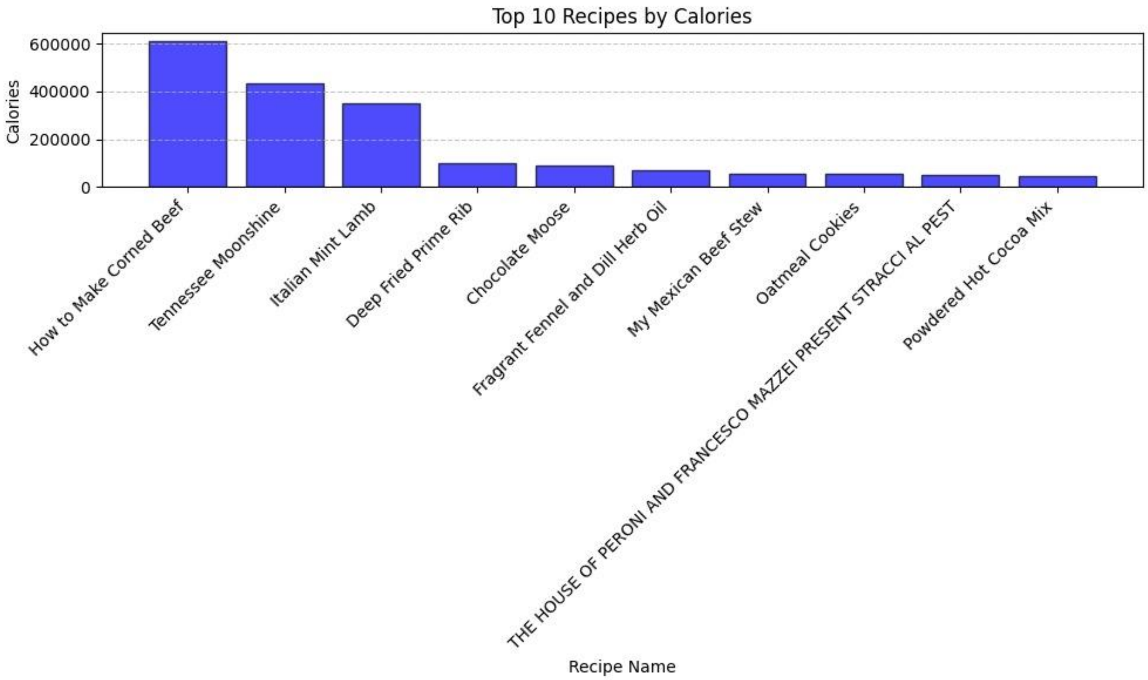
Methods

There were several layers to the meal plan tool; the primary functions we wanted the tool to be able to accomplish were:

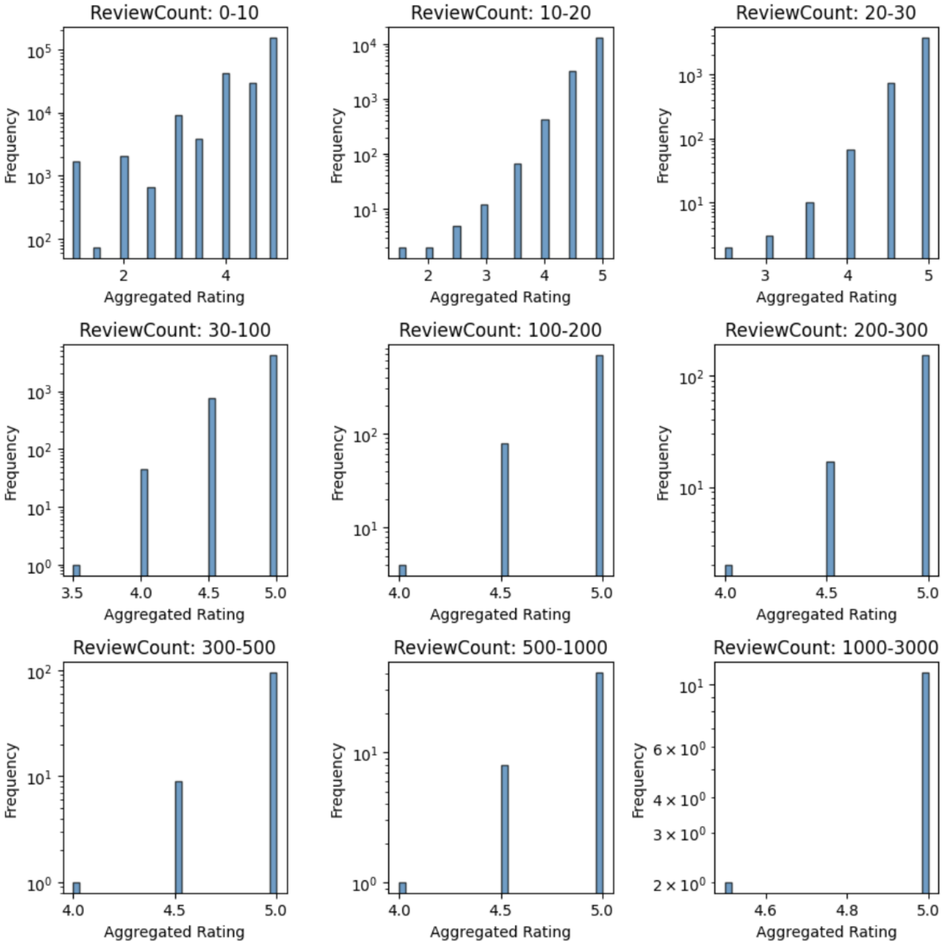
1. Filter down to meaningful recipes
2. Analyze the reviews and ratings for the recipes and determine “true” meal quality
3. Analyze the reviews and recipes to determine the similarity of recipe clusters (minimizing similarity results in maximized variety)

4. Classify the meal types to ensure that a breakfast meal was not recommended for dinner (or vice versa)
5. Ensuring that the total meal plan satisfied the set nutritional requirements, with an attempt to balance for each day (within acceptable tolerance)

The data set used for this project was a database of recipes (and corresponding reviews) for approximately 500k meals extracted from food.com in 2020. The data set was not ideal – there were some recipes for nonsense/joke meals (such as a recipe for leave-in conditioner); recipes for meals with no proper portion/serving size; recipes with excessive calories per serving (> 2000 calories per serving); and recipes for meals missing some or all nutritional information. Additionally, there were recipes that had no reviews. Each of these recipes were filtered out of the database through simple null-value elimination or removing the obvious anomalies mentioned above, leaving us with a database of approximately 240,000 recipes with over a million reviews.



Once the data set was properly filtered, it became apparent that the number of 5 star ratings overwhelmed the lower ratings. If all ratings are set to max, it can be extremely difficult to choose one recipe over another. We felt that these 5-star reviews were not always genuine – like a 5-star rating in Uber, where it's the "default" rating so long as nothing terrible happens. The next goal was to "normalize" the ratings.

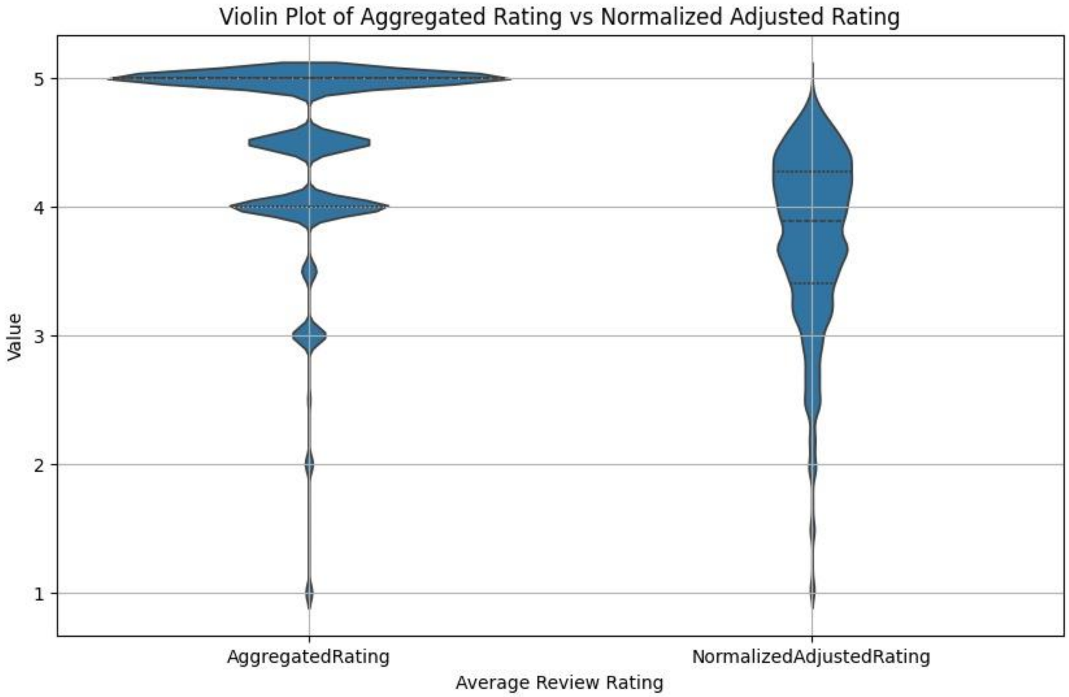


Logarithmic histogram of ratings distribution among recipes (sorted by review count)

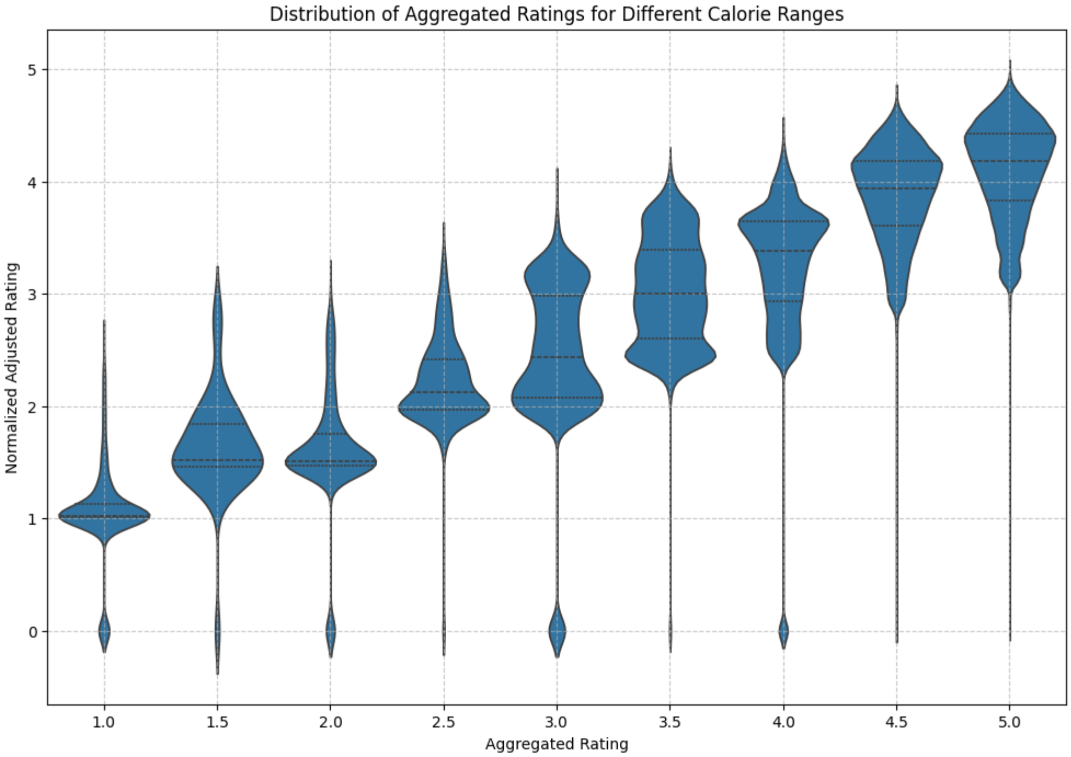
To correct for this, we weighted the ratings based off the original weighting score itself (50%), the “popularity” (25%) which was the summed sentiment of the text reviews given for each recipe, and the average sentiment of reviews for each recipe (25%). The adjusted rating (r_a) is simply a weighted average of each rating score based on the weight (w) and the different adjusted ratings (r_w).

$$r_a = \sum w r_w$$

Additionally, the summed and average sentiment scores were uniformly distributed through quartile-based normalization based on their respective sentiment before combining for the weighted score. Sentiment scores were calculated using a pre-trained model that specializes in cross-domain sentiment analysis (Zhou et al., 2020). After a cursory review of the sentiment scores, it seemed to perform well with more glowing reviews receiving a better score. One can see how this method preserves most of the information as they are strongly positively correlated with each other, however the adjusted ratings have a much broader distribution. This allowed the ratings to be something more interesting and useful to optimize on, because the ratings are no longer so biased towards class imbalance.



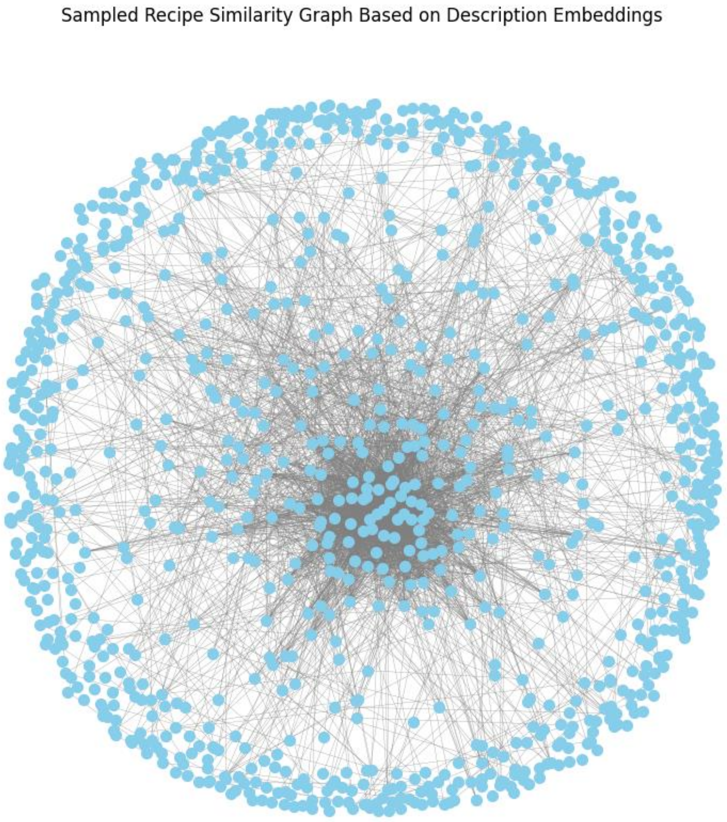
Violin distribution of original (left) and adjusted (right) ratings

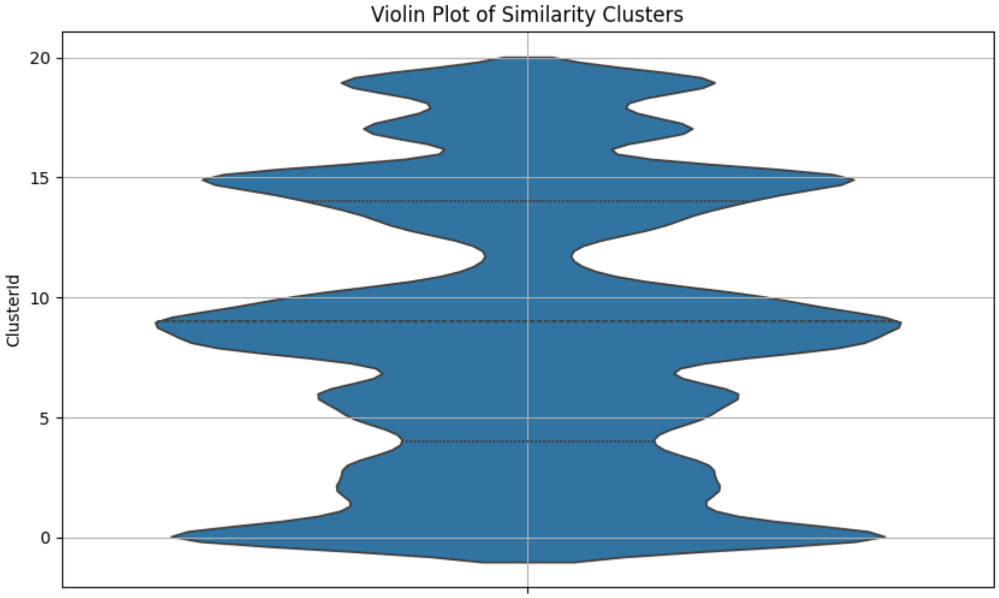


Violin distribution of adjusted rating separated by various ranges of original rating

The above graphs show the new distribution of ratings after the sentiment rating, as well as a breakdown by range of original rating. We felt that the second visual greatly validated our normalized rating, because as the original average rating of the recipes dropped, the center of distribution of our adjusted rating dropped – we could basically draw a straight line through the centers of distribution from left to right and bottom to top. This showed that our adjustments were not overcorrecting and kept in line with the general ratings for the recipes.

To ensure the meal plan contained a variety of meals, we decided to measure their similarity. A pre-trained sentence transformer model was used to generate text embeddings for the recipe descriptions and keywords. We then implemented a k-means clustering model to cluster similar recipes into 20 groups that we used as a proxy for types of food. This allowed us to add diversity in the meal plan by calculating pairwise hamming distance of the clusters, which represent diversity of the similar clusters between recipes, and using that as another optimization objective.


Sampled Recipe Similarity Graph Based on Description Embeddings

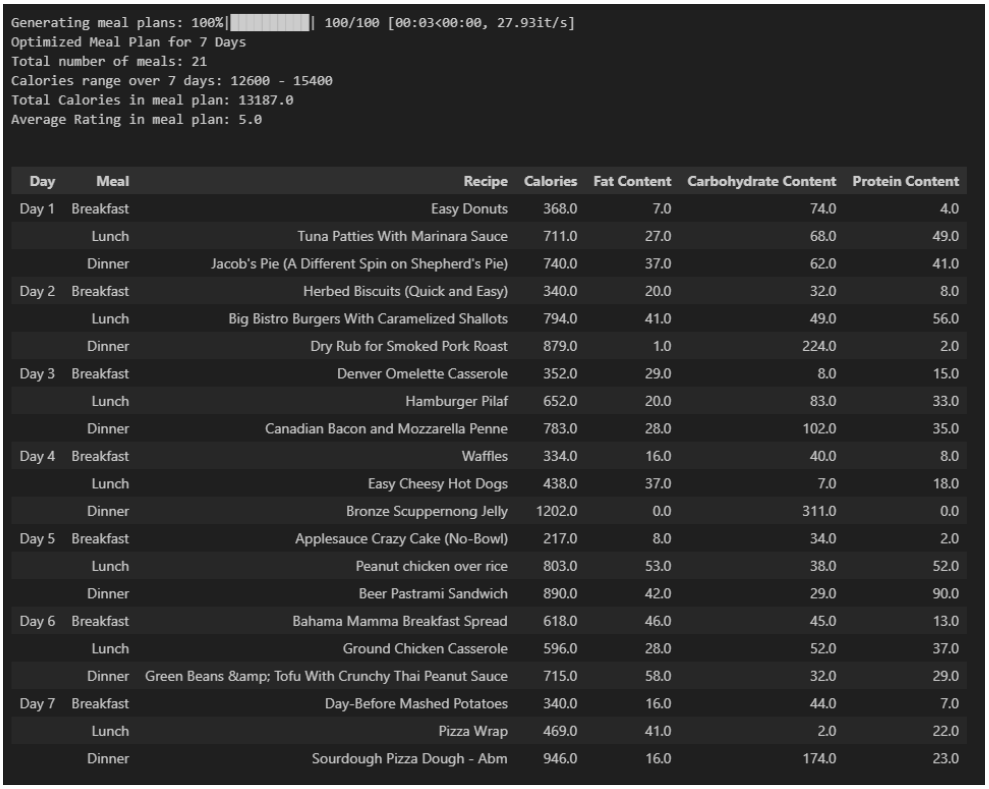


We also decided to classify each meal into breakfast, lunch, or dinner to create a realistic and enjoyable meal schedule. Although around 60% of the filtered recipe descriptions had specific references to when in the day they should be eaten, the rest of the foods had no such specifications. To classify the rest of the meals, we used the same text embeddings from each meal's description, along with keywords, and used them to train a logistic regression model. By using a training and validation split of 0.3, we were able to increase the test accuracy to 95%. Once trained, we were able to use the model to classify all meals as breakfast, lunch, or dinner. Finally, with the data pre-processed, it could finally be sent to the optimization model for meal scheduling.

For our constrained discrete optimization, we use a linear programming model to generate multiple meal plans optimized for adjusted ratings and constrained by nutritional content. We use equality constraints to set the number of total meals along with the number of breakfasts, lunches, and dinners. We use inequality constraints to set limits for nutritional metrics (calories, fat, carbs, protein) and diversity (based on numbers of each cluster from the similarity analysis). We set our bounds to 0 or 1 for all recipes so that they would be included or excluded. We then solve to maximize our adjusted ratings. Once we have a set of meals that fits our nutritional requirements, we order them so that each day has one of each meal type and pass this solution to a list of meal plans that will be further optimized for diversity. Once we generate a desired number of meal plans, we calculate diversity through a pairwise hamming distance calculation of the ClusterIDs in the meal plan which represent similarity. We then select the meal plan that has the most diversity.

Results and Discussion

The model was successful on all accounts. We were able to generate 100 meal plans that are constrained by our nutritional needs and optimized for taste (adjusted rating), then select the most diverse plan, all in < 5 seconds.



```
Generating meal plans: 100%|██████████| 100/100 [00:03<00:00, 27.93it/s]
Optimized Meal Plan for 7 Days
Total number of meals: 21
Calories range over 7 days: 12600 - 15400
Total Calories in meal plan: 13187.0
Average Rating in meal plan: 5.0
```

Day	Meal	Recipe	Calories	Fat Content	Carbohydrate Content	Protein Content
Day 1	Breakfast	Easy Donuts	368.0	7.0	74.0	4.0
	Lunch	Tuna Patties With Marinara Sauce	711.0	27.0	68.0	49.0
	Dinner	Jacob's Pie (A Different Spin on Shepherd's Pie)	740.0	37.0	62.0	41.0
Day 2	Breakfast	Herbed Biscuits (Quick and Easy)	340.0	20.0	32.0	8.0
	Lunch	Big Bistro Burgers With Caramelized Shallots	794.0	41.0	49.0	56.0
	Dinner	Dry Rub for Smoked Pork Roast	879.0	1.0	224.0	2.0
Day 3	Breakfast	Denver Omelette Casserole	352.0	29.0	8.0	15.0
	Lunch	Hamburger Pilaf	652.0	20.0	83.0	33.0
	Dinner	Canadian Bacon and Mozzarella Penne	783.0	28.0	102.0	35.0
Day 4	Breakfast	Waffles	334.0	16.0	40.0	8.0
	Lunch	Easy Cheesy Hot Dogs	438.0	37.0	7.0	18.0
	Dinner	Bronze Scuppernong Jelly	1202.0	0.0	311.0	0.0
Day 5	Breakfast	Applesauce Crazy Cake (No-Bowl)	217.0	8.0	34.0	2.0
	Lunch	Peanut chicken over rice	803.0	53.0	38.0	52.0
	Dinner	Beer Pastrami Sandwich	890.0	42.0	29.0	90.0
Day 6	Breakfast	Bahama Mamma Breakfast Spread	618.0	46.0	45.0	13.0
	Lunch	Ground Chicken Casserole	596.0	28.0	52.0	37.0
	Dinner	Green Beans &amp; Tofu With Crunchy Thai Peanut Sauce	715.0	58.0	32.0	29.0
Day 7	Breakfast	Day-Before Mashed Potatoes	340.0	16.0	44.0	7.0
	Lunch	Pizza Wrap	469.0	41.0	2.0	22.0
	Dinner	Sourdough Pizza Dough - Abm	946.0	16.0	174.0	23.0

With the limited timeframe of the project, we were not able to exhaustively filter recipes and validate meal types, but you can see from the generated meal plan above, that not only do the nutritional values fit the constraints, the categorization mostly fits as well.

We believe that much of this work is generalizable and scalable to other engineering projects. First, data analysis is always an important step. Most projects involving large sources of data deal with some sort of lapse in data control, in need of preprocessing. Even after filtering the dataset for several reasons (null or anomalous values for key metrics), there was still a need to clean the data. It always depends on the use case, sometimes you can impute the data. In our case, we used a regression model to help classify meal types for many anomalous entries. This is transferrable to many other engineering tasks where we can use machine learning to improve imputation tasks beyond simply filling down or interpolating. We also often have problems with data that is disproportionately distributed. We can use machine learning to help us utilize a meaningful spread of data, in a similar manner to how we redistributed ratings. Using machine learning to add other meaningful features to our datasets is also a fairly universal application. In our case we processed text data to provide similarity scores for diversity analysis. It is common to process data for projects in multiple ways to better understand correlations between features.

Conclusion

In conclusion, we believe that our project can provide a meaningful solution to people that struggle with aligning their nutritional goals with their culinary enjoyment. Through the creation of a machine learning enabled tool, we constrain meal plans based on a user's specific nutritional goals and restrictions as well as optimize for taste and diversity to provide a meal plan that is not only nutritious but also easier to follow. We also believe that the methods and tools that we used here would be applicable to other engineering design efforts, especially the clarification of large amounts of data that seems to be overly similar.

Team Contributions

Everyone contributed to the project in different ways. James and Bobby had a larger role in Python coding our different subsystems. Stephen made more of an impact working on the poster and report. We all collaborated on how we could use machine learning to improve our constrained discrete optimization. We were satisfied with each other's work and each delivered on our obligations to the project deliverables.

Bublitz, M. G., Peracchio, L. A., & Block, L. G. (2010). Why did I eat that? Perspectives on food decision making and dietary restraint. *Journal of Consumer Psychology*, *20*(3), 239–258. https://doi.org/https://doi.org/10.1016/j.jcps.2010.06.008

Mela, D. J. (1999). Food choice and intake: the human factor. *Proceedings of the Nutrition Society*, *58*(3), 513–521. https://doi.org/DOI: 10.1017/S0029665199000683

Zhou, J., Tian, J., Wang, R., Wu, Y., Xiao, W., & He, L. (2020). Sentix: A sentiment-aware pre-trained model for cross-domain sentiment analysis. *Proceedings of the 28th International Conference on Computational Linguistics*, 568–579.