SustainaMeal: AI-Powered Recipe Recommendations for Sustainable Eating

Giovanni Tempesta
g.tempesta16@studenti.uniba.it

Michele Di Carlo
m.dicarlo6@studenti.uniba.it

February 7, 2024

Abstract

This project try to address the critical need for sustainable and healthy eating habits in a world increasingly conscious of environmental impact and personal health. By analyzing a comprehensive dataset of recipes, we focus on identifying alternative culinary choices that offer improved sustainability and health benefits. The goal is to guide individuals towards making informed food choices that positively affect both personal well-being and environmental sustainability, addressing key concerns in contemporary dietary practices.

1 Introduction

The intersection of **dietary habits** with personal health and environmental sustainability forms the foundation of our project. In a world increasingly attuned to the impacts of food choices, this work aims to guide individuals towards more **sustainable and health-conscious culinary alternatives**. Recognizing the pivotal role of technology in achieving this, we have developed a comprehensive system integrating various advanced tools and methodologies.

We started our journey from a **dataset of recipes** which is the aggregation of a diverse sourced from the web. Utilizing **data processing techniques**, we embarked on a detailed examination of recipes, trying to compute the **carbon footprint (CFP)** and **water footprint (WFP)** of individual ingredients. This granular analysis enabled us to assess the overall sustainability of each recipe, paving the way for informed recommendations.

Central to our project is the development of a **custom-built library**, tailored to suggest alternative recipes that strike a balance between health benefits and minimal environmental impact.

The culmination of our efforts is manifested in **'SustainaMeal'**, a specialized library developed to reach our goals. SustainaMeal leverages the combined capabilities of our dataset analysis,transformers, custom function, and a primordial conversational agent to offer users a novel approach to recipe selection, one that is deeply rooted in the principles of sustainability and health consciousness.

Further enhancing the user engagement, we have integrated in our system **LangChain**, employing it to create a primordial **conversational agent**. This agent serves as the interface between our library and the users, facilitating an interactive and dynamic recipe discovery experience.

The following sections of this document delve deeper into each aspect of our work. We discuss the existing literature in **Related Work**, elucidate our data sources and methodologies in **Dataset** and **Preprocessing**, detail the functionalities of **SustainaMeal**, and explore the experiments done on the library. Finally, we conclude with reflections on our achievements and potential future directions in the **Conclusion and Future Works** section.

2 Related Work

The work titled "FoodPrintDB: an extensive database for recipes sustainability estimation" was conducted by Gigantelli Alberto and Iacovazzi Antonio Raffaele [**?**]. Its primary aim was to create a comprehensive database containing information about ingredients and recipes, intended for future developments in food recommendation systems.

The main focus of this work was on ensuring high-quality information and utilizing reliable sources. Specifically, significant attention was given to constructing a database that incorporates meaningful data regarding sustainability. This enabled the provision of sustainable alternatives for specific recipes. The parameters considered to evaluate sustainability were the Carbon Foot Print and the Water Foot Print.

The Carbon Foot Print denotes the total amount of carbon dioxide emissions produced during the production, transportation, and use of a given product or ingredient. On the other hand, the Water Foot Print measures the impact of water usage on water resources.

The first version of the database was originally released by two other colleagues, Matteo Fusillo and Salvatore Amoruso, as support for their recommendation system. Subsequently, additional information from the SU-EATABLE Life database was integrated into this initial version. In particular, the Carbon Foot Print and Water

Foot Print values were added, contributing to the development of the second version of the FoodPrintDB.

A sustainability calculation was also carried out in order to have a concrete value to represent how sustainable the recipe is compared to others.

These previous works proved fundamental in recovering the information on wfp and cfp necessary to calculate the sustainability and healthiness levels of our library.

3 Dataset

3.1 Recipes

3.1.1 Data Origin

The recipes in our dataset were obtained through a comprehensive web scraping process. This involved systematically collecting data from various online culinary websites and recipe databases.

3.1.2 Data Structure

The dataset comprises a total of 37 columns, among which the following have been identified as most critical for our analysis:

- **recipe_id**: A unique identifier for each recipe.
- **title**: The name of the recipe.
- **ingredients**: List of ingredients used in the recipe.
- **tags**: Categorization tags associated with each recipe.
- **calories [cal]**: The total calorie content of the recipe.
- **caloriesFromFat [cal]**: The amount of calories derived from fat.
- **totalFat [g]**: Total fat content in grams.
- **saturatedFat [g]**: Amount of saturated fat in grams.
- **cholesterol [mg]**: Cholesterol content in milligrams.
- **sodium [mg]**: Sodium content in milligrams.
- **totalCarbohydrate [g]**: Total carbohydrates in grams.
- **dietaryFiber [g]**: Dietary fiber content in grams.

- **sugars [g]**: Total sugars in grams.
- **protein [g]**: Protein content in grams.
- **who_score**: A healthiness score for each recipe based on the World Health Organization (WHO) methodology, ranging from 0 to 14, with 14 being the best.
- **fsa_score**: A healthiness score based on the UK Food Standards Agency (FSA) nutrient profiling system, ranging from 0 to 8, with 8 being the best.
- **nutri_score**: A nutritional score for each recipe, graded from A (best) to E (worst).

This structure allows for a comprehensive analysis of each recipe's nutritional content, healthiness, and overall profile, facilitating our objective of suggesting healthier and more sustainable culinary alternatives.

Initially, the dataset contains 507,335 recipes. However, this number is reduced through preprocessing to ensure data quality and relevance. A detailed discussion of the preprocessing steps and their impact on the dataset will be provided in Chapter 4.

The dataset includes a total of 902 unique tags, which serve as categorization labels for the recipes. These tags are instrumental in the SustainaMeal library for filtering and suggesting recipes based on the original recipe. The most significant tags, based on the frequency of occurrence and relevance to our objectives, are:

- **Main dish**: 71,285 recipes
- **Desserts**: 42,807 recipes
- **Breakfast**: 13,384 recipes
- **Appetizers**: 20,187 recipes
- **Vegetables**: 53,485 recipes
- **Meat**: 50,740 recipes
- **Seafood**: 14,722 recipes
- **Vegetarian**: 35,599 recipes
- **Fruit**: 31,245 recipes
- **Pasta rice and grains**: 23,924 recipes

- **Chicken**: 20,304 recipes
- **Pork**: 12,664 recipes

These tags play a crucial role in the functionality of SustainaMeal, enabling the system to provide tailored recipe suggestions. A more detailed exploration of how these tags are utilized within SustainaMeal will be discussed in Chapter 5.

3.1.3 Limitations

One significant limitation of the dataset is the presence of excessive noise in the ingredients data. This issue arises due to the unstructured nature of web-sourced recipe data, where ingredients are often listed in varying formats and with different levels of detail. For instance, ingredients might be described with brand names, specific preparation styles, or non-standard measurements, leading to inconsistencies and challenges.

This noise can impact the accuracy of our analysis, particularly when assessing the nutritional content and sustainability of the recipes. Identifying and standardizing these diverse ingredient descriptions is a complex task, requiring sophisticated data cleaning and preprocessing methods. The strategies employed to address this challenge and mitigate its impact on our analysis are detailed in Chapter 4, dedicated to data preprocessing.

3.2 CSEL

3.2.1 Data Origin

The dataset comes from an elaboration of the SU-EATABLE Life(SEL) database, which is a multilevel database on the carbon (CF) and water footprint (WF) values of food raw materials.

3.2.2 Data Structure

The dataset is composed of 6 columns, that can be described in this way:

- **Food Commodity GROUP**: The general category to which food goods belong
- **Food commodity ITEM**: Specific ingredient within the category
- **Food commodity TYPOLOGY**: subcategory within a food group, particular typology

- **Food commodity sub-TYPOLOGY**: Subcategory of the subcategory within a food group
- **final co2**: Final value of Carbon Foot Print
- **final wfp**: Final value of Water Foot Print

3.2.3 Limitations

One of the limitations, to the previously described dataset, pertains to the presence of noise within the data. Specifically, ingredient names contain additional words or characters (such as the asterisk (*)). Another limitation is the dataset's limited coverage of ingredients, totaling a mere 471 items. This problem restricts the availability of data necessary for computing the sustainability of certain recipes, necessitating the exploration of alternative approaches. Furthermore, an additional constraint arises from the absence of commonly found ingredients within our dataset, such as salt. To address these challenges, we've resorted to employing methodologies outlined in the data preprocessing section.

4 Preprocessing

It proved necessary to carry out preprocessing in order to reduce the discrepancy between the ingredients of the CSEL dataset and that of the recipe dataset. In particular we carried out the following steps:

4.1 Recipes Dataset

Recipe Dataset Reduction Previously, sustainability calculation involved matching ingredients from all the recipes with those from the CSEL dataset. In this updated approach, focus shifts to 'usable' recipes:

- A recipe is considered usable if:
  - It has no duplicates (recipes with identical titles are dropped).
  - It possesses tags (recipes with 'tags' property as null are dropped).
  - It contains at least one ingredient (recipes with 'ingredients' property as null are dropped).

This step reduced recipes from 507,335 to 214,800.

Recipe Dataset Cleaning Removal of adjectives, stopwords, verbs, numbers, special characters, and content within parentheses.

4.2 CSEL Dataset Cleaning

- Removal of ingredients lacking both cfp and wfp values.
- Elimination of adjectives, stopwords, and special characters like asterisks (*).
- Removal of all parentheses and their contents, e.g., '(F)', '(G)', etc.

This step focuses on key terms and reduces noise in ingredient texts, aiding in matching. Following the previous recipe reduction, ingredients to match decreased from 230,620 to 159,284.

4.3 Ingredient Matching

- Checking if the CSEL dataset ingredient name is contained within our ingredient name.
- Checking if the ingredient name is contained within the CSEL ingredient name.

This matching process assigned 78,998 out of 159,284 ingredients.

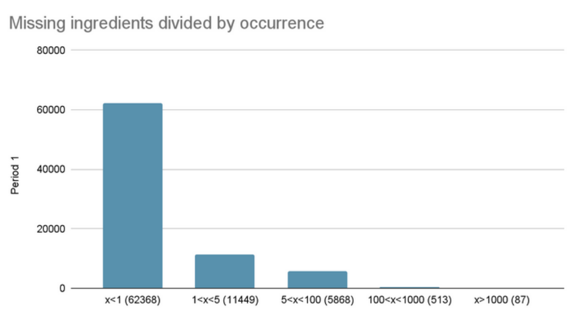

Figure 1: Missing Ingredients divided by occurrence.

Transformer Usage for Missing Ingredients

- Utilized a transformer to calculate similarity between missing ingredients (occurring more than 1 time) and matched ingredients.
- Set a threshold of 0.98 as the minimum similarity.
- Obtained around 240 possible similarities, manually reviewing and eliminating the inconsistent ones, resulting in 178 similarities.

Manual Assignment of Most Relevant Missing Ingredients After automated analysis, further manual intervention addressed 87 missing ingredients with more than 1000 occurrences, finding 19 possible associations.

Creation of Dictionary Created a dictionary containing names of all processed ingredients with their respective cfp and wfp values.

Recipes dataset reduction We employ the same approach as the previous work 'Food Print DB' for managing recipes with missing ingredients. Thus, we proceed to reduce our dataset by eliminating all recipes that we do not deem valid. A recipe is considered valid when:

- The percentage of known and valid ingredients is greater than a specified threshold.
- An ingredient is considered valid if we have the values for both cfp and wfp

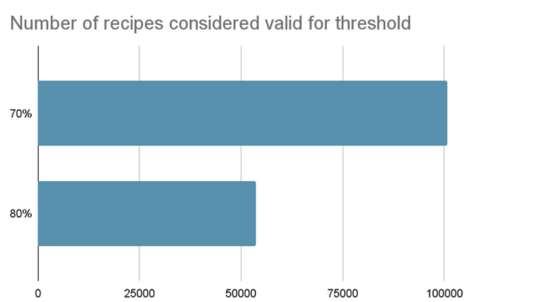

Figure 2: Number of recipes considered valid for threshold.

Currently we have decided to use recipes that respect the threshold constraint of 70% (100870 out of 214800). Therefore the final number of recipes is 100870.

4.4 Sustainability Calculation Procedure

Subsequently, the 'calculate dss score recipe' function was utilized to execute the actual sustainability calculation using the provided recipe index. The ISS (Ingredient Sustainability Score) was computed for each ingredient within the recipe employing the 'iss score' function. The ISS calculation followed the formula:

$$\text{ISS} = \alpha \times \text{Ncfp}(x) + \beta \times \text{Nwfp}(x)$$

Here, α is set to 0.8 and β to 0.2, based on outcomes from previous experiments conducted by the formula developers. $\mathrm{Ncfp}(x)$ represents the normalized Carbon Foot Print value, and $\mathrm{Nwfp}(x)$ denotes the normalized Water Foot Print value for the ingredient.

After obtaining the ISS for each ingredient within the recipe, the DSS (Recipe Sustainability Score) was calculated using the formula:

$$\mathrm{DSS} = \sum_{i=0}^{|K|-1} \mathrm{ISS}(K_i)e^i$$

Following this, a set K was constructed containing the ingredients associated with a recipe, sorted in descending order based on their ISS. Here, K_0 represents the least sustainable ingredient, and $K_{|K|-1}$ denotes the most sustainable one. $\mathrm{ISS}(K_i)$ signifies the ISS value of the ingredient, and 'e' is the numerical constant valued at 2.71. This process facilitated the compilation of the list of DSS for each recipe, constituting an array of recipe sustainability indices.

Finally, the ultimate sustainability score was computed using the formula:

$$\text{Sustainability Score} = \frac{\text{DSS(R) - MinDSS}}{\text{MaxDss - MinDss}}$$

This comprehensive process enabled the derivation of a normalized sustainability index for each recipe that closely resembled a given one, providing an assessment of overall sustainability based on the employed ingredients.

Label Assignment Employing a similar methodology as in the previous study, we assign labels to each recipe based on the computed score. Recipes are categorized into three labels:

- Low (score >= 0.5): Indicating recipes with low sustainability.
- Medium (0.1 < score < 0.5): Representing moderately sustainable recipes.
- High (score <= 0.1): Indicating highly sustainable recipes.

Within our dataset, we represent these labels using integers:

- 0 = High (16,433 recipes)
- 1 = Medium (79,157 recipes)
- 2 = Low (5,280 recipes)

5 SustainaMeal

5.1 Introduction

SustainaMeal is a Python library designed to suggest alternative recipes for healthier or more sustainable options. Leveraging machine learning and natural language processing, it compares nutritional profiles and semantic similarities to provide recipe recommendations.

5.2 Architecture

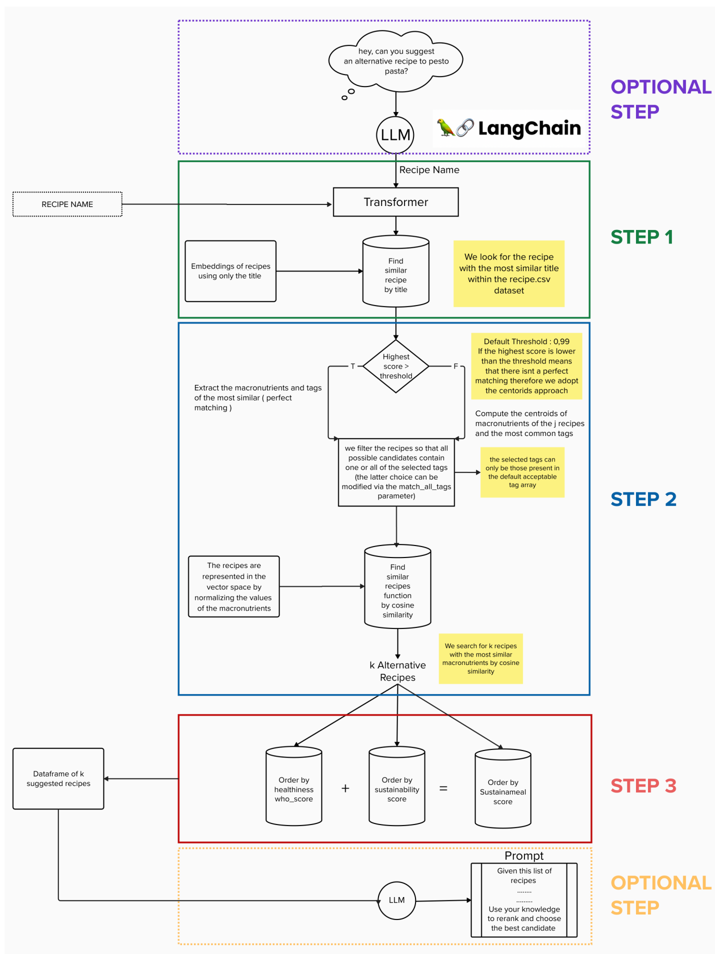

Figure 3: SustainaMeal Architecture.

5.2.1 Step 1: Recipe Title Embedding and Similarity Search

The first step in the SustainaMeal library involves processing the recipe name through a transformer-based model to generate embeddings. These embeddings, which capture the semantic essence of the recipe titles, serve as the basis for identifying similar recipes within the dataset. The system compares the embedding of the input recipe title to the embeddings of titles in the recipe.csv dataset, aiming to find the recipe with the closest match.

5.2.2 Step 2: Nutritional Similarity and Tag Filtering

In the second step, the SustainaMeal library extracts the macronutrients and **tags** from the most similar (perfectly matching) recipe obtained from Step 1. It then filters the recipes to ensure that all potential candidates contain at least one or more of the selected tags. Recipes are represented in a vector space by normalizing the values of the macronutrients. A cosine similarity function is used to find similar recipes, searching for k alternative recipes with the closest macronutrient profile. If the highest similarity score is below a default threshold (0.99), indicating no perfect match, the system adopts a c**entroid approach** to compute the macronutrient profile of the k (k is the number of candidates provided as parameter of the function) recipes and their **most common tags**.

5.2.3 Step 3: Ranking by Healthiness , Sustainability or SustainaMeal score

The final step within the SustainaMeal library involves ranking the filtered recipes according to their healthiness , sustainability or both. This is accomplished by sorting the recipes based on two distinct scores: the ‘who_score‘, which assesses the healthiness based on the World Health Organization guidelines, or the sustainability label, which evaluates the environmental impact of the recipes. To combine the two rankings we created a new score called Sustainameal score which tries to consider both previous scores when reordering the recipes.

$$\text{sustainameal\_score} = \text{sustainability\_score}(*) \cdot \alpha + \text{who\_score} \cdot \beta \tag{1}$$

(*) sustainability_score –> because the sustainability score is inverted (in the initialization phase) to be compatible with the who_score since in the original score the higher the sustainability score, the more unsustainable the recipe. Alpha and Beta are the weights which by deafult are 0.7 and 0.3.

5.2.4 Optional step: Reranking using LLM

The final step within the SustainaMeal library involves ranking the filtered recipes according to their healthiness or sustainability. This is accomplished by sorting the recipes based on two distinct scores: the ‘who_score‘, which assesses the healthiness based on the World Health Organization guidelines, or the sustainability label, which evaluates the environmental impact of the recipes. This dual-ranking system ensures that users are recommended recipes that not only fit their nutritional requirements but also align with environmental sustainability practices.

5.2.5 Optional Step: Agent initialization and recipe extraction

Step 0 of the library is an optional phase designed to bootstrap a rudimentary conversational agent leveraging the gpt3.5-turbo model. We’ve engineered a bespoke tool named "AlternativeSustainableRecipeTool," utilizing the langchain framework, which is adept at identifying the name of a recipe a user seeks alternatives for. Following the retrieval of the recipe name, subsequent steps involve invoking specific functions aimed at generating recommendations through the conventional workflow of the library. It’s noteworthy that the library remains functional even in the absence of this initial step. The default response mechanism employs a reranking function powered by gpt3.5.

LangChain is a versatile framework designed to augment the capabilities of conversational agents. It integrates seamlessly with advanced language models, enabling agents to access and leverage a wide array of external knowledge sources and language processing tools. This integration allows for the creation of sophisticated and contextually aware interactions, pushing the boundaries of what conversational agents can achieve. In the SustainaMeal project, LangChain plays a pivotal role in bridging the gap between user questions and actionable functions to get the recipe recommendations.

5.2.6 Step 5: LLM ReRank

As an optional last step, it is possible to utilize GPT-3.5-turbo to reorder the recipes obtained from the previous steps. This allows the use of Large Language Models (LLMs) for selecting the best recipe. This approach leverages the advanced natural language understanding capabilities of GPT-3.5-turbo to assess and prioritize the recipes based on various criteria, potentially including healthiness, sustainability, ingredient preferences, or any other factors deemed important. This step enhances the decision-making process by integrating the nuanced comprehension and evaluative skills of GPT-3.5-turbo into the selection of the optimal recipe.

5.3 Functionalities

5.3.1 Initialization

Initialization of the SustainaMeal system, including data loading and preparation of transformer embeddings.

```
def __init__(self, recipes_df, nutrients,load=False
    transformer_name='davanstrien/autotrain-recipes-2451975973'):
```

5.3.2 Find Similar Recipes

A function to find recipes similar to a given input based on title similarity and nutritional profile.

```
def find_similar_recipes(self, input_text, k, acceptable_tags
            , match_all_tags,check_sustainability=False):
```

5.3.3 Order by Healthiness

This function orders recipes by healthiness score, considering WHO guidelines.

```
def order_recipe_by_healthiness(self, nearest_recipes=None,
    score='who_score'):
```

5.3.4 Order by Sustainability

Orders recipes based on a sustainability score, reflecting environmental impact considerations.

```
def order_recipe_by_sustainability(self, nearest_recipes=None,
        score='sustainability_label', secondary_sort_field='who_score'):
```

5.3.5 Order by SustainaMeal

Orders recipes based on our custom score SustainaMeal.

```
def order_recipe_by_sustainameal(self, nearest_recipes=None,
                                 alpha=0.7, beta=0.3):
```

5.3.6 Gpt ReRank

We have developed a new method that fully entrusts the task of ranking alternative recipes to GPT during its use. Essentially, we have shifted the entire responsibility of ranking to GPT to later compare the order we implemented with that obtained from GPT. This function utilizes the GPT-3.5 Large Language Model (LLM) from OpenAI to select the best recipe from a list, ordered based on sustainability and healthiness criteria. It constructs a prompt for GPT-3.5, asking it to rank the recipes from most recommended to least recommended.

```
{
Prompt used:
Using your knowledge, please rank (if necessary) the following
recipes from most to least recommended based on a balance
of sustainability and healthiness:

1. Recipe:Healthy Salad
2. Recipe: Quinoa Bowl
3. Recipe: Veggie Stir-Fry
}
Which one should I choose? Return just the name.
```

the defintion of the functions involved :

```
def setup_openai_key(self, open_ai_key):
def choose_best_recipe_with_gpt(self, nearest_recipes=None,
                                 alpha=0.7, beta=0.3):
```

5.3.7 Agent

As previously introduced, within our library, we have implemented a conversational agent using the LangChain framework. Through this framework, it is possible to create and train one's own agent according to specific rules, allowing for particular responses to certain questions, with the option to integrate a Large Language Model (LLM) as support. Specifically, our agent has been trained to determine the healthiest and most sustainable alternative recipe.

The workflow of our agent is structured as follows:

1. **Catch the Recipe:**
   - Takes as input the recipe that we want to analyze.
2. **Find Similar Recipes:**
   - Utilizes the implemented methods for searching for the most similar alternatives, compiling a list of these recipes.
3. **SustainaMeal Sorting:**
   - Sorts the considered recipes based on the SustainaMeal score.
4. **Rerank with GPT:**
   - Utilizes GPT as an LLM to reorder the aforementioned recipes.

This approach enables the agent to provide recommendations for alternative recipes based on health and sustainability criteria, integrating the analysis of similar alternatives, SustainaMeal scores, and GPT's reranking capabilities to achieve optimized results.

```
def create_agent(self):
def agent_ask(self, text):
```

5.3.8 CMD script

We have implemented the integration of our library to enable direct execution from the terminal. For this purpose, we created a file named 'clip.py', which contains the complete code to handle commands. This file is linked to the library through the 'setup.py' file. However, we noticed that each time a command is executed from the terminal, it initiates a new session. This means that the embedding of titles, which is the main limitation in terms of time, requiring about 20 minutes to complete, is processed again with each execution. To improve efficiency, we decided to save the processed data, thus avoiding the need to repeat the embedding process with each start. This significantly reduces processing time and enhances the overall user experience when using the library from the terminal.

6 Experiments

We conducted experiments on four distinct types of recipes:

- 100 recipes with a WHO score above average.
- 100 recipes labeled as 1 (moderately sustainable).
- 100 recipes labeled as 2 (highly unsustainable).
- 30 unknown recipes (recipes not present in the recipe dataset used)

The experiments combined different values for nutrients, represented in two configurations:

```
Configuration A:
Calories [cal] , Total Fat [g], Sodium [mg] ,
Dietary Fiber [g] , Sugars [g], Protein [g]

Configuration B:
Calories [cal] , Total Fat [g] , Saturated Fat [g] ,
Cholesterol [mg] , Sodium [mg] , Dietary Fiber [g]
Sugars [g] , Protein [g]
```

Values considered for k:

- 1
- 10
- 50
- 100

Options for matching all tags:

- True
- False

The tables below represent the average increase or decrease (in percentage) of the top 10 recipes (except for k=1) in:

- H = healthiness (who_score)
- S = sustainability ($sustainability\_score$)
- SM = sustenameal_score (the new score implemented with $a = 0.7$ and $b = 0.3$)

for all the recipes of each subgroup with every possible combination of parameters.

6.1 Experiments with known recipes

6.1.1 Scenario 1 : Moderately Healthy Recipes

For moderately healthy recipes we considered the recipes that have the healthiness score (who_score) above the average .In the current dataset, given the reduction of the recipes, the values for who_score are between a maximum of 0.5 and a minimum of 0. Consequently we calculated that the average value was 0.215 and considered the recipes above this value.Furthermore, by already working on recipes considered "healthy", it becomes more difficult for the bookshop to find healthy alternatives.

```
{"title":"Avocado Crab Crostini",
"who_score":0.2725611984,
......
}
```

	k	nutrients	match_all_tags	H_inc_mean	S_inc_mean	SM_inc_mean
0	1	B	False	-0.61	2.38	1.70
1	1	B	True	-0.33	1.17	0.75
2	1	A	False	-2.94	0.47	-0.22
3	1	A	True	-2.49	1.04	0.23
4	10	B	False	-2.20	1.42	0.55
5	10	B	True	-2.07	0.11	-0.50
6	10	A	False	-3.82	-0.75	-1.49
7	10	A	True	-3.54	-1.13	-1.74
8	50	B	False	3.41	17.72	15.27
9	50	B	True	4.27	16.60	14.44
10	50	A	False	3.19	17.66	15.20
11	50	A	True	3.58	17.13	14.78
12	100	B	False	6.38	19.92	17.53
13	100	B	True	6.94	19.34	17.08
14	100	A	False	5.51	20.19	17.67
15	100	A	True	6.08	19.50	17.12

Figure 4: Moderatly healthy recipes results table

In the figure 4, it is observable that when focusing on recipes with healthiness score above the average, there are decreases in terms of healthiness when considering a lower k value, specifically k < 10. Conversely, when increasing the value of k, such as k = 50 or k = 100, there are increments observed. This is due to the fact that we increase the number of potential candidates, so there is subsequently a greater likelihood of observing increments.

6.1.2 Scenario 2 : Moderately Sustainable Recipes

For moderately sustainable recipes we consider the recipes that have the sustainability label equals to 1 (the initial computed score is between 0.1 and 0.5)

Example:

```
{"title":"Roasted Pepper and Artichoke Panini",
"sustainability_score":0.3682138075,
"sustainability_label":1,
......
}
```

	k	nutrients	match_all_tags	H_inc_mean	S_inc_mean	SM_inc_mean
0	1	B	False	-4.08	0.05	-0.52
1	1	B	True	-4.69	1.67	0.85
2	1	A	False	1.16	0.67	0.10
3	1	A	True	2.02	1.90	1.26
4	10	B	False	18.29	2.93	2.34
5	10	B	True	18.41	3.51	2.82
6	10	A	False	7.86	1.69	1.13
7	10	A	True	7.39	2.43	1.72
8	50	B	False	50.94	18.90	17.55
9	50	B	True	52.40	18.96	17.72
10	50	A	False	52.49	19.68	18.33
11	50	A	True	52.88	19.14	17.87
12	100	B	False	64.54	21.98	20.70
13	100	B	True	66.81	21.93	20.90
14	100	A	False	69.27	22.70	21.38
15	100	A	True	70.76	22.29	21.18

Figure 5: On average sustainable recipes results table

In the figure 5, while focusing on recipes of moderate sustainability, increments in sustainability are observed across all parameter configurations except for some configurations with k=1. Naturally, when discussing recipes of moderate sustainability, the increments are not substantial but nonetheless valid.

6.1.3 Scenario 3 : Unsustainable Recipes

For Unsustainable recipes we consider the recipes that have the sustainability label equals to 2 (the initial computed score is above 0.5)

```
{"title":"Easy Lasagna - No Ricotta",
"sustainability_score":0.6594095694,
"sustainability_label":2
......
}
```

	k	nutrients	match_all_tags	H_inc_mean	S_inc_mean	SM_inc_mean
0	1	B	False	2.77	68.41	54.27
1	1	B	True	1.11	42.35	33.05
2	1	A	False	2.76	78.42	62.05
3	1	A	True	0.36	42.33	32.85
4	10	B	False	2.44	67.28	53.53
5	10	B	True	5.03	41.54	33.02
6	10	A	False	4.19	74.69	59.55
7	10	A	True	3.97	44.34	34.87
8	50	B	False	8.04	127.85	103.21
9	50	B	True	9.08	112.34	90.24
10	50	A	False	8.47	132.48	106.90
11	50	A	True	8.97	115.75	92.97
12	100	B	False	10.84	137.63	111.52
13	100	B	True	13.43	130.75	105.69
14	100	A	False	12.70	139.03	112.89
15	100	A	True	12.47	132.46	106.86

Figure 6: Highly unsustainable recipes results table

In the figure 6, employing solely unsustainable recipes, we will consistently experience increments in terms of sustainability across all conceivable parameter configurations. This consistent observation underscores that, under varying parameter settings, the system consistently achieves enhancements in sustainability when dealing exclusively with unsustainable recipes.

6.2 Experiments with unknown recipes

	k	nutrients	match_all_tags	H_inc_mean	S_inc_mean	SM_inc_mean
0	1	B	False	-3.00	2.58	1.83
1	1	B	True	-2.91	-0.20	-0.56
2	1	A	False	-3.00	2.58	1.83
3	1	A	True	2.64	0.46	0.76
4	10	B	False	2.96	-3.25	-2.69
5	10	B	True	3.93	-1.43	-0.98
6	10	A	False	1.51	-4.88	-4.36
7	10	A	True	3.20	-4.56	-3.85
8	50	B	False	13.12	14.95	14.47
9	50	B	True	13.79	14.93	14.54
10	50	A	False	12.98	14.84	14.50
11	50	A	True	12.67	15.07	14.66
12	100	B	False	15.77	18.18	17.64
13	100	B	True	17.66	18.06	17.76
14	100	A	False	16.43	17.87	17.51
15	100	A	True	17.78	17.90	17.71

Figure 7: Unknown recipes results table

In the figure 7, when considering recipes not present in our dataset, with k=1, there are decreases in terms of healthiness in almost all combinations, unlike sustainability, except the row with nutrient of type A and match_all_tags setted to True. Only in the row with nutrients of type B and match_all_tags setted to True, we have a decrement in sustainability. Increasing k, for example with k=10, shows improvements in healthiness in all combinations, while in this case, unlike before, there are decreases in sustainability in all instances.

With k=50 and k=100, there are increases in both healthiness and sustainability in all cases. These findings indicate how the choice of the k parameter influences the evaluations of healthiness and sustainability of recipes.

6.3 Best configuration

From all the experiments conducted in the aforementioned scenario, it became evident that the choice of k is crucial for increasing the average improvement in healthiness and sustainability, while the selection of nutrients and tags affects the similarity (at the ingredient level) between recipes. Essentially, changing the match_all_tags parameter to true or false only alters the degree of freedom in the choice of possible candidates but does not vary the increase in healthiness and sustainability. Below is what we consider to be the best combination:

```
k=50
nutrients = [Calories [cal] , Total Fat [g], Sodium [mg],
            Dietary Fiber [g] , Sugars [g], Protein [g]]

tags = ['appetizers', 'main-dish', 'side-dishes', 'drinks',
        'beverages', 'fruits', 'desserts','breakfast',
        'pasta-rice-and-grains', 'rice', 'pasta','pizza',
        'breads', 'meat', 'fish','seafood', 'beef',
        'chicken', 'vegetarian']

match_all_tags = false
```

6.4 Experimentation Gpt ReRank vs No ReRank Comparison

The following table 8 illustrates the average increases in metrics for the top-ranked recipe. We conducted two sets of comparisons: the first set with $k = 10$ for both (to assess the extent of improvement lost when GPT is asked to rerank), and the second set comparing the results obtained without reranking at $k = 1$ against reranking with $k = 10$. Specifically, the initial row pertains to the attributes of the highest-ranked recipe selected by GPT from among ten candidate recipes. The subsequent row corresponds to the attributes of the top-ranking recipe derived from our ranking algorithm applied to the same pool of ten candidate recipes. Finally, the last row relates to the singular recipe identified as possible substite to a given reference recipe.

Top_H_inc_mean	Top_S_inc_mean	Top_SM_inc_mean	gpt_rerank	k
3.26	71.33	56.07	True	10
6.22	129.08	103.87	False	10
2.77	68.41	54.27	False	1

Figure 8: Example Reranking comparison

The data in the figure 8 that applying GPT reranking with $k = 10$ results in a significant increase in the average healthiness , sustainability scores and the sustainameal score, but certainly lower than the average increases of the top 1 recipes of the scenarios where reranking is not used. In contrast, without reranking at $k = 1$, the average increases for healthiness and sustainability are lower than when reranking with $k = 10$, suggesting that as we expected, reducing the number of candidates is not a recommended practice and to choose a recipe to return as input it is preferable to set a k > 1 and ask gpt to perform the rerank. The main advantage of gpt (as we will see in the section 6.5) would seem to be the ability to choose the recipe most consistent with the input recipe (without knowing the name of the input recipe).

6.5 Qualitative Analyses

To assess the ability to chose the best recipe of the reranking procedure,in the re-ranking experiments, we have set the "match_all_tags" flag to false. This configuration enhances the variety of recipe categories, making the search less restrictive. Subsequently, we delegate the ranking task to GPT, which tends to designate as the "best recipe" the one that is both healthier and more sustainable while maintaining coherence with the input recipe.

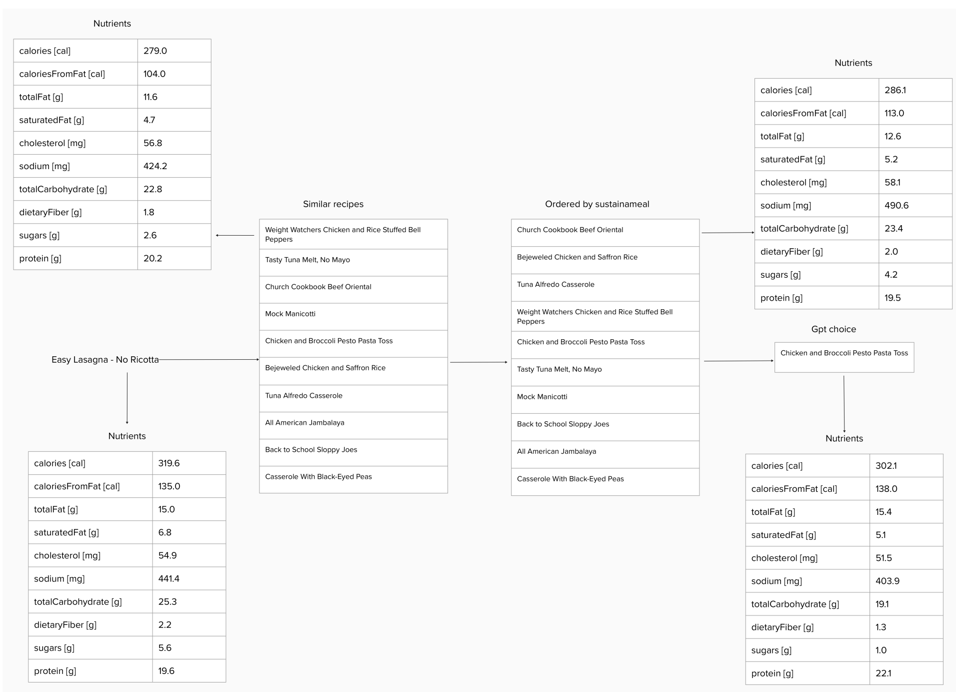

Figure 9: Results for Easy Lasagna - No Ricotta

In this initial example of figure 9, we consider the recipe 'Easy Lasagna - No Ricotta'. We calculate, using our method, the most similar recipes. From the obtained results, we have viable alternatives such as 'Chicken and Broccoli Pesto Pasta Toss' and 'Tasty Tuna, No Mayo'. Subsequently, upon sorting based on the Sustainameal score, our top choice is the 'Church Cookbook Beef Oriental', followed by 'Bejeweled Chicken and Saffron Rice', and lastly, 'Tuna Alfredo Casserole'. Finally, from these recipes, we delegate the ranking task to the GPT-3.5 Turbo language model, which classifies 'Chicken and Broccoli Pasta Pesto Toss' as the best recipe. From these results, it is evident that while our ranking method prioritizes a meat-based recipe as the top choice, GPT, in contrast, designates a pasta-based recipe as the best choice. This aligns more coherently with the input recipe.

Figure 10: Results for Spicy Homemade Hamburger Helper

In this second example in the figure 10, we consider the recipe 'Spicy Homemade Hamburger Helper' as the input. Through our method, we calculate the most similar recipes. From the results obtained, notable and healthier alternatives include 'Fillet Bubby Blake,' 'Tasty 2-step Chicken,' and 'Lemon Pepper Chicken Stew.' Subsequently, upon sorting using the Sustainameal score, the top-ranked recipe is 'Round Steak With Beer Marinade for the Grill,' followed by 'Chinese Pork With Shallots' in the second position, and 'Korean Fire Pork-Toeji Kogi Kui' securing the third spot. Finally, by delegating the ranking of all similar recipes to the GPT-3.5 Turbo language model, we determine 'Chinese Pork With Shallots' as the best recipe.

6.6 Exploring Experimental Results Online

To facilitate the exploration of our experimental data, we have developed interactive HTML and JavaScript pages. These pages provide a user-friendly interface for navigating through the dataset, allowing users to intuitively understand and analyze the results of our experiments. The visualization tools implemented on these pages enable filtering, sorting, and detailed examination of specific data points within the broader context of our proposal.

These interactive pages are hosted on GitHub Pages, ensuring easy access and broad availability. They serve not only as a means of presenting our findings but also as a platform for collaborative exploration, where users can contribute their insights or identify patterns that may not have been initially apparent.

The data navigation tools can be accessed through the following GitHub Repository link:

`https://github.com/GiovTemp/SustainaMeal_Case_Study`

7 Conclusion and Future Works

In conclusion, the approach developed in this project has proven to be quite promising. Our primary objective was to create a system capable of providing sustainable and healthy alternatives to a given recipe, aiming to promote the use of environmentally friendly foods for both environmental conservation and personal health. The system demonstrates its ability to offer valid alternatives, showing a substantial percentage increase.

For future developments in this project, some promising directions could be considered:

1. Improve dataset preprocessing with a more refined data cleaning process to achieve even more accurate and reliable results.
2. Implement a query engine to justify the results obtained by the system, enhancing transparency and understanding of the decision-making process.
3. Explore further insights into alternative research, including new algorithmic approaches or the integration of additional data sources to further improve the diversity and quality of generated proposals.

These perspectives could contribute to consolidating and refining the system, allowing for greater effectiveness and adaptability to the needs of users interested in promoting a sustainable and healthy lifestyle.

References

[1] Salvatore Amoruso, Matteo Fusillo, A. (2022). *FoodPrint: a web app for a sustainable lifestyle*. Salvatore Amoruso. Matteo Fusillo.