FoodPrintDB: an extensive database for recipes sustainability estimation

Department of Computer Science
Master's Degree in Computer Science

Semantics in Intelligent Systems
A.Y. 2022/2023

Professors:
Semeraro Giovanni
Musto Cataldo

Students:
Gigantelli Alberto
Iacovazzi Antonio Raffaele

Introduction

Today's world is facing several challenges against problems like climate change, biodiversity loss, and poverty. As a response to these issues, in 2015 the United Nations established 17 Sustainable Development Goals (SGD)[1] that should be reached by 2030 for the benefit of humankind. Among these goals, there are number 2, which aims to allow fair access to good quality food for all, and number 13, which aims to diminish the impact of greenhouse gasses that cause global warming. To achieve these two goals, it is important to nudge a sociocultural change in the way we produce and consume food, limiting food waste and preferring foods that are more healthy and sustainable in terms of carbon emissions and water use.

A good way to motivate people and help them to pursue a change in terms of dietary habits is to give them suggestions about sustainable recipes using tools like recommender systems since these kinds of systems could be easily integrated into everyday life[2]. In order to make this possible, it is necessary to have a good dataset that can represent the properties of a wide range of ingredients and the recipes obtainable using them. In particular, it is crucial to have information about the carbon footprint and the water footprint of ingredients, in order to make it possible to estimate the environmental impact of a recipe.

The aim of this work is to build a database of ingredients and recipes that can be used as a foundation for future work on a food recommender system. In particular, the focus is on the quality of information and the use of trustworthy sources. This will be done starting from an already existing database called FoodPrintDB and trying to overcome its limitation expanding the number of information available.

Our hope is that this work could be a step toward the SGD goals and help future research on this fundamental topic.

Food and sustainability: a complex problem

“**Global health**” is an evolving concept that considers a person's health as physical, mental, and social well-being within the global context influenced by socioeconomic, political, socio-demographic, legal, and environmental factors. This has made it one of the cornerstones of the (SDG) Sustainable development Goals and requires the concept of sustainable development to preserve human health and the environment.[3.1]

The threat to global health [3.2] is caused by changes in food and production patterns due to industrialization, globalization, urbanization, and increased incomes, leading to a homogenization of global diets with higher fat, sugar, and salt content and causing the spread of excess malnutrition and non-communicable diseases. Malnutrition is becoming a global epidemic causing both *overweight and obesity* as well as *chronic deprivation of food* and *hidden hunger*[3.3] for micronutrients. Intensive food production also has a negative impact on the environment, causing global warming, catastrophic climatic events, reduced agricultural production, and increased malnutrition. Food production contributes significantly to greenhouse gas emissions, accounting for 30% of global emissions.

All this leads to the creation of several paradoxes, among which we recognize 3 relevant[3.4]:

1. Extreme malnutrition (and the opposite) despite sufficient food production for all of humanity.
2. About ⅓ of food production is used to feed livestock and to produce fuel, in the face of a scarcity of resources and growing food insecurity.
3. Approximately ⅓ of the world's food production is wasted each year, i.e. more than a billion tons of food.

From the point of view of production and consumption, however, we encounter two phenomena called *Food loss* and *Food waste*, which occur respectively during production, post-harvest, processing, or intentional waste by consumers. Food loss is more present in developing countries while food waste is in rich countries [3.5].

Carbon footprint (CFP) and water footprint (WFP)

The carbon footprint (CFP) refers to the total amount of carbon dioxide emissions produced as a result of the manufacturing, transportation, and use of a product or ingredient. These emissions can come from sources such as fuel combustion for energy production, and deforestation for agriculture and food processing.

The water footprint (WFP), on the other hand, measures the impact of water usage on the water resource. This includes the amount of water required for growing and manufacturing an ingredient, as well as water used during food processing and processing. This impact can have negative consequences on the availability of drinking water and on the health of aquatic ecosystems, especially in areas where water resources are scarce.

These two terms can be used to calculate a recipe's sustainability score as an indicator of the carbon emissions and water impact of the individual ingredients included in the recipe. The sum of the scores of the individual ingredients in a recipe could be used to obtain an overall score that represents its ecological footprint[4] and it can be used to promote more conscious and sustainable food choices.

Challenges in gathering information on recipe sustainability

Building a reliable knowledge base about food sustainability is not simple for several reasons, the main ones are

1. Location: Ingredient availability and cooking practices vary by location, this means that there may be farm-to-table products that reference the use of local, seasonal produce to prepare meals or products that come from far, this makes it difficult to gather information on sustainable recipes in a given geographical area.
2. Correlation with quantities: the quantity of an ingredient needed for a recipe can vary due to many factors, such as personal taste, ingredient availability, and local habits[5]. This makes it difficult to establish the true sustainability of an ingredient or recipe, as a less sustainable ingredient may be used in small quantities and vice versa.
3. Diversity and sub-categories: There are many sub-categories of sustainable recipes, such as vegetarian, vegan, gluten-free recipes, etc., making it difficult to collect information covering all categories.
4. Inability to fully trace: Tracing the impact of ingredients in the supply chain can be difficult or even impossible, making it difficult to determine the sustainability of a recipe with certainty.

In conclusion, the sustainability of the food system is an important factor for human health and the environment. Lifestyles and dietary habits have a significant impact on health and life expectancy, and creating a carbon and water footprint dataset for food commodities is an important step in understanding and monitoring the environmental impact of food supply chains. This information can help prevent climate change and support the long-term sustainability of food supply chains. Detailed knowledge of the quantities of resources used and the emissions generated for the production of different food raw materials is essential to identify opportunities for improvement and support research on food sustainability.

FoodPrintDB_v1: properties and issues

The first version of FoodPrintDB (FoodPrintDB_v1) derives from the work of Matteo Fusillo and Salvatore Amoruso as support for their food recommender system. The database was realized by combining different sources like the *Recipe1M+* and the *CORGIS* datasets, the *myemission.com* and *healabel.com* websites data, and the *Edamam* API as a tool for combining all the information and fetching more data about ingredients.

The database provided the following characteristics:

- 133 food categories extracted from CORGIS.
- 352 ingredients were obtained joining the information in the Recipe1M+ dataset with the ones stored on the Edamam website which also allows fetching data about macronutrients. 135 over 352 ingredients present information about carbon footprint fetched by myemission.com or by means of data mining techniques. 72 over 352 ingredients present information about water footprint fetched by healabel.com or still derived using data mining. 214 over 352 ingredients have information about the relative category.
- 51235 recipes taken from Recipe1M+ and related to the proper ingredients using the Edamam API, 357 are the ones whose at least 80% of ingredients own information about carbon and water footprints.

This version of the database presents several problems, first of all, the recipes whose has enough information about ingredients sustainability are really few making this database not so useful for sustainability tasks, furthermore, data about carbon and water footprint are taken from resources whose sources are not public and the data mining process used to infer further data could have been introduced some biases in data.

In order to overcome these problems it is necessary to refer to a source that has solid academic reliability and lists a wide range of ingredients as well as a way to link this information to the ingredients present in the FoodPrintDB. After some research, the SU-EATABLE database was chosen as a reference.

SU-EATABLE Life database description

SU-EATABLE Life (SEL) database is a multilevel database of carbon (CF) and water (WF) footprint values of food commodities, based on a standardized methodology to extract information and assign optimal footprint values and uncertainties to food items, starting from peer-reviewed articles and gray literature [6]. It has been realized as support for the European project SU-EATABLE LIFE which aims to develop policies and cultural changes to reduce the environmental impact of food both in production and consumption. The database is available as an xls file.

The SEL database was realized following three steps.

In step 1 a massive massive review of academic and public information about food footprint was done in order to collect as much data as possible.

The principal sources of this collection are the work of Clune et al [7], the Double Pyramid Database [8] besides other works found using SCOPUS and the Google Scholar engine.

After data collection, all the food *items* were assigned to a specific group, typology, and sub-typology and then related to additional information like the source and the publication year.

Finally, the collected CF values were further analyzed and handled in order to obtain the system boundary harmonization using the approach reported by Clune et al [7].

Before going on, it is necessary to further explain what groups, typology, sub-typology, and items are in this context.

- A **Group** is a wide category of food commodities. The database includes 4 groups: Agricultural processed, Animal husbandry, Crops, and Fishing
- A **Typology** is an aggregated level of food commodity as generally known in the food system. It represents a group of items having similar characteristics. For example, the typology “legumes” includes peas, lentils, beans, soybeans, etc.
- A **Sub-typology** is used to subdivide three typologies (vegetables, fruit, shellfish) which include a large variability of items (i.e. the typology “shellfish” includes the sub-typologies crustacean, bivalves, and cephalopods).
- An **Item** is the most detailed level for aggregated data. Generally, the item name corresponds to common market definitions (ex. tomato, mussels, milk). The level of detail proposed depends on the available data source.

In step 2 all the data collected in the previous step are aggregated according to the 4 levels of aggregation.

This analysis was done using statistical methods that estimate the most reliable value for every item, in fact, if certain conditions are met the median value of the items with the same name in the collection can be used as CF/WF value for that item, otherwise, the median value of the relative typology or sub-typology is suggested for representing the item.

In step 3 all the data gathered and manipulated in the previous steps are then rearranged so that final users can easily access and understand them. In the SEL database, there are two distinct sheets called “SEL for CF users” and “SEL for WF users”, both sheets represent data using the same data structure and are composed of the following fields (some fields change name when referring to CF or WF):

- **Food commodity group**: contains the commodity group of the ingredient.
- **Food commodity item**: contain the actual name of a specific food item. Some ingredients can additionally report information about the fact they are imported or grown in

greenhouses using special marks. More information about these marks can be found in the "INTRO" sheet of the database.

- **Carbon Footprint kg CO2eq/kg or l of food ITEM | Water Footprint liters water/kg o liter of food ITEM**: represent the information about the carbon or water footprint computed using the median value for that item.
- **Uncertainty**: this field represents how much the information collected for that item is statistically reliable.
- **Suggested CF value/Suggested WF value**: tell which value is suggested as a reliable CF/WF value of that item among the available ones.
- **Food commodity typology**: represents an aggregation level of a category of similar foods based on their characteristics.
- **Carbon Footprint g CO2eq/g o cc of food typology | Water Footprint cc water/g o cc of food typology**: represents the information about the carbon or water footprint of the item's typology.
- **Food commodity sub-typology**: it is a level of aggregation lower than that of the typology, which is used to divide the broader food categories into specific subgroups. This information is available only for some specific food items.
- **Carbon Footprint g CO2eq/g o cc of food sub-typology | Water Footprint cc water/g o cc of food sub-typology**: represents the information about the carbon or water footprint of the item's sub-typology, if available.

Food Commodity Group	Carbon footprint			Water footprint		
	Items	Typology	Sub-typology	Items	Typology	Sub-typology
Agri. processed	100 (570)	41	—	95 (328)	33	—
Animal husb.	46 (1530)	21	—	67 (308)	22	—
Crops	117 (986)	18	8	136 (55)	15	8
Fishing	60 (263)	5	3	22 (55)	2	1
Total	**323 (3349)**	**85**	**11**	**320 (937)**	**72**	**9**

To sum up, this table represents the number of CF and WF data of food commodities reported in the SEL database at different informative levels as items, typologies, and sub-typologies. Their breakdown into the 4 food commodity groups is reported. In brackets are reported the total number of data entries used to calculate the food item values. For further details on the data reported in this table and on the SEL database in general, it is advisable to consult the original paper [6].

In the end, this database is highly flexible since it is delivered along with a clear methodological framework that allows further expansion. For example, when it is necessary to assign a CF or a WF value for an unknown food item it's possible to refer to the median CF/WF value related to the typology or sub-typology compatible with that item. This point is beneficial for the aims of this work since it allows us to determine CF/WF value even for the ingredients not represented in the SEL database without loose scientific reliability.

Using SEL Database as the source for FoodPrintDB_v1 completion

As we have seen the SEL Database covers several kinds of foods and is delivered with scientifically accurate information about cfp and wfp, due to these characteristics it plays an ideal role as a reference for the construction of databases in the food and sustainability domain. In this section are explained all the performed steps to migrate the useful information of the SEL Database into FoodPrintDB_v1.

SEL database preprocessing

Despite its solid methodological framework, the SEL database is not provided in a way that allows us to easily use it, in fact for every ingredient there are multiple possible choices of CF/WF value that must be chosen accordingly with the indication presented in the “Suggested CF/WF value” field, furthermore, there are two distinct lists of ingredients, one about the carbon footprint and the other one about the water footprint. The ingredients listed are not the same since they were derived from different studies, so it is necessary to merge the two lists and fill possible gaps.

The “SEL for CF users” and “SEL for WF users” sheets are the main sources used in the context of this work since they are easily manageable and clear in their contents.

Step 1: selection of the right CF/WF value

After the conversion in .csv of the two sheets, two new Pandas data frames were created using a simpler data structure that lists only the name of the ingredient (food item) along with the hierarchical information (group, typology, and sub-typology) and a single cfp/wfp value.

This value was selected following what is stated in the “Suggested CF/WF value” field of the original csv file. In the case of multiple choices available between the different values, the most specific one was preferred.

The two data frames obtained were saved in csv format.

Step 2: merge the two sheets

This step has the aim of building a single csv file with both the information about cfp and wfp.

The merge was done starting from the cfp csv file and following different strategies.

First of all for every food item in the data frame, a food item with the same name was searched in the wfp csv file removing the strings “(I)”, “(G)”, “*” and “**” both sides, these characters represent specific properties of the food item where:

- “(I)” represents an imported food item.
- “(G)” represents a food item grown and heated in a greenhouse.
- “(g)” represents a food item grown in a greenhouse but not heated.
- “*” means that an item corresponds to the typology or sub-typology, i.e. that no other items are included in that typology of food, which however describes a broader group of food items.
- “**” means that the median of an item is calculated using CF median of subgroups, i.e. computing first the median value of each subgroup and then, using these intermediate values, the final value for the item is computed.

This is because on the water footprint side, there isn’t the concept of imported or grown in greenhouses foods, and the presence of that kind of mark in the cfp ingredient name could spoil some good matches, furthermore, given a food item in both the cfp and the wfp list, is not granted that the asterisk character appears in both the version, so it must be removed.

There is also the string “(F)” that represents frozen food items, but we choose to not remove this information in this step since we thought that a frozen food item should be directly mapped only with another frozen food item.

For every ingredient matched, the value of its relative wfp was reported in a new column called “final_wfp”.

This trivial strategy allows us to cover 153 ingredients both with cfp and wfp. For the remaining ingredients, it was necessary to use as wfp value the wfp related to the typology (or sub-typology if available) of each ingredient.

The typology/sub-typology matching was performed by removing possible notes in parentheses on both sides since we noticed that the same information could be reported with slightly different syntax in the two different csv, using this technique we reached a total of 251 matches.

We missed some matches because there were ingredients in the cfp csv such that the related typology was a more detailed version of a similar typology in the wfp csv. To cover these ingredients as well, a manual conversion between typologies was made following the idea that a more specific typology can be mapped onto a more general one.

The list of mappings is reported in the following table:

CFP uncovered typology	**WFP compatible typology**	**Note**
CHEESE HARD & SEMI-HARD	CHEESE	
CHOCOLATE	COCOA DERIVATES	
FISH FROZEN	FISH	
FISH PROCESSED	FISH	
FRUIT JUICE	JUICE	
JUICE LOCAL	JUICE	

FRUIT FROZEN	FRUIT	
FRUIT HEATED GREENHOUSE	FRUIT	
FRUIT IMPORTED	FRUIT	
FRUIT NOT HEATED GREENHOUSE	FRUIT	
FRUIT JUICE IMPORTED	JUICE	
LEGUMES NOT HEATED GREENHOUSE	LEGUMES FRESH	All legumes without wfp were green beans that are fresh legumes.
LEGUMES CANNED	LEGUMES FRESH	
LEGUMES FROZEN	LEGUMES FRESH	
LEGUMES	LEGUMES FRESH	
NUTS	NUTS SHELLED	Since the shelled product is more common for the final user in supermarkets, however, this choice is debatable.
SHELLFISH FROZEN	SHELLFISH	
VEGETABLES HEATED GREENHOUSE	VEGETABLES	

Using these mapping the typologies in the cfp data frame were changed placing the compatible one of the wfp data frame, then performing again the matching between typologies we reach a total of 302 ingredients of the cfp data frame covered with the information about water footprint.

In order to expand the number of available ingredients the same overall procedure was performed also on the wfp csv taking into account only the ingredients which didn't already appear in the cfp csv.

The steps performed are basically the same, also here was necessary to find a proper matching between different, but compatible, typologies.

The table of these mappings is the following:

WFP uncovered typology	CFP compatible typology	Note
COCOA DERIVATIVES	CHOCOLATE	Chocolate is more specific, but chocolate is the one closer to it having cfp information; could be a debatable match.
STARCHY TUBER	STARCHY TUBERS	Same name, just different syntax.
JUICE	FRUIT JUICE	Same typology, just different syntax.
PROCESSED GRAINS	GRAINS	

CHEESE	CHEESE HARD & SEMI-HARD	Cheese hard is more specific but the actual items involved can be mapped onto it.
EGGS & DERIVATIVES	EGGS	The item involved are eggs part
NUTS SHELLED	NUTS	
NUTS WITH SHELL	NUTS	
LEGUMES FRESH	LEGUMES	
LEGUMES DRIED	LEGUMES	
COFFEE	COFFEE GROUND & PARCHMENT	Since the item involved is coffee beans

The overall matching procedure between wfp csv exclusive ingredients and cfp typologies allows us to add further 147 ingredients into the final dataset.

The items in the wfp csv for which wasn't possible to have a cfp typology match were excluded since our priority was to have information about cfp, more important in terms of pollution estimation of an ingredient, however, this choice can be easily modified in order to add also those ingredients into the dataset.

The obtained dataset contains a total of 472 ingredients covered by cfp information whereas 449 have also wfp information.

The dataset was saved using the csv format and restoring the original syntax of ingredients, typology, and sub-typology in order to not lose any information. This is a choice open to change for future works since a more compact and standard syntax could be more useful.

Integration of SEL database data into FoodPrintDB: FoodPrintDB_v2

After generating a compact and comprehensive version of the SEL database (from now will be called CSEL) the next step was to link the information into the FoodPrintDB_v1 creating the new FoodPrintDB_v2.

The main idea is to understand which food item of CSEL is more relatable for each ingredient of the original FoodPrintDB, in case no ingredient in the CSEL was appropriate then a compatible typology with the relative cfp/wfp will be used.

In order to do so we used the transformers-based model *all-MiniLM-L6-v2* freely available on Hugginface[11] to compute the similarity between each ingredient name in the FoodPrintDB and each food item in the CSEL dataset.

For each ingredient, the top 2 matches were given as output along with the information about the relative cfp, wfp, and food item name of that ingredient delivered as sql update, then each match was manually checked in order to understand if it was reliable or not then. For the ones considered reliable, the relative sql update was saved. We choose to save also the CSEL food item name storing it in a new table called “ingredients_name_alias” in order to have a simpler string that identifies each ingredient, this could be useful for nlp related algorithms where the original names of ingredients in the FoodPrintDB could be confusing.

For all the ingredients which weren't matchable with any food item in the CSEL we opted to use a similar algorithm to find a proper match with the typology names and the relative carbon footprint and water footprint using the two sheet "SEL CF Typologies STAT" and "SEL WF Typologies STAT" previously converted in csv files.

Here the best-matched typology was given as output along with the median value of the cfp/wfp delivered as a sql update. Then for each ingredient, the proper match was manually selected (if present) and finally, a proper alias name was manually derived from the original ingredient's name.

After these operations, for some ingredients, still there wasn't a match with any food item or typology. For each of these ingredients, a proper typology was selected manually and used as a reference for the matching algorithm in order to "force" the right match. All the so-obtained matches are delivered with a handmade name alias for each ingredient.

Details about the nature of the matches are reported in an xls file called "Mapping_foodprintDB_sueatable" which describes, using a color code, the origin of each match along with its relative food item or typologies matched. This document could be also a starting point to train algorithms with the aim of automatically detecting matches in similar situations.

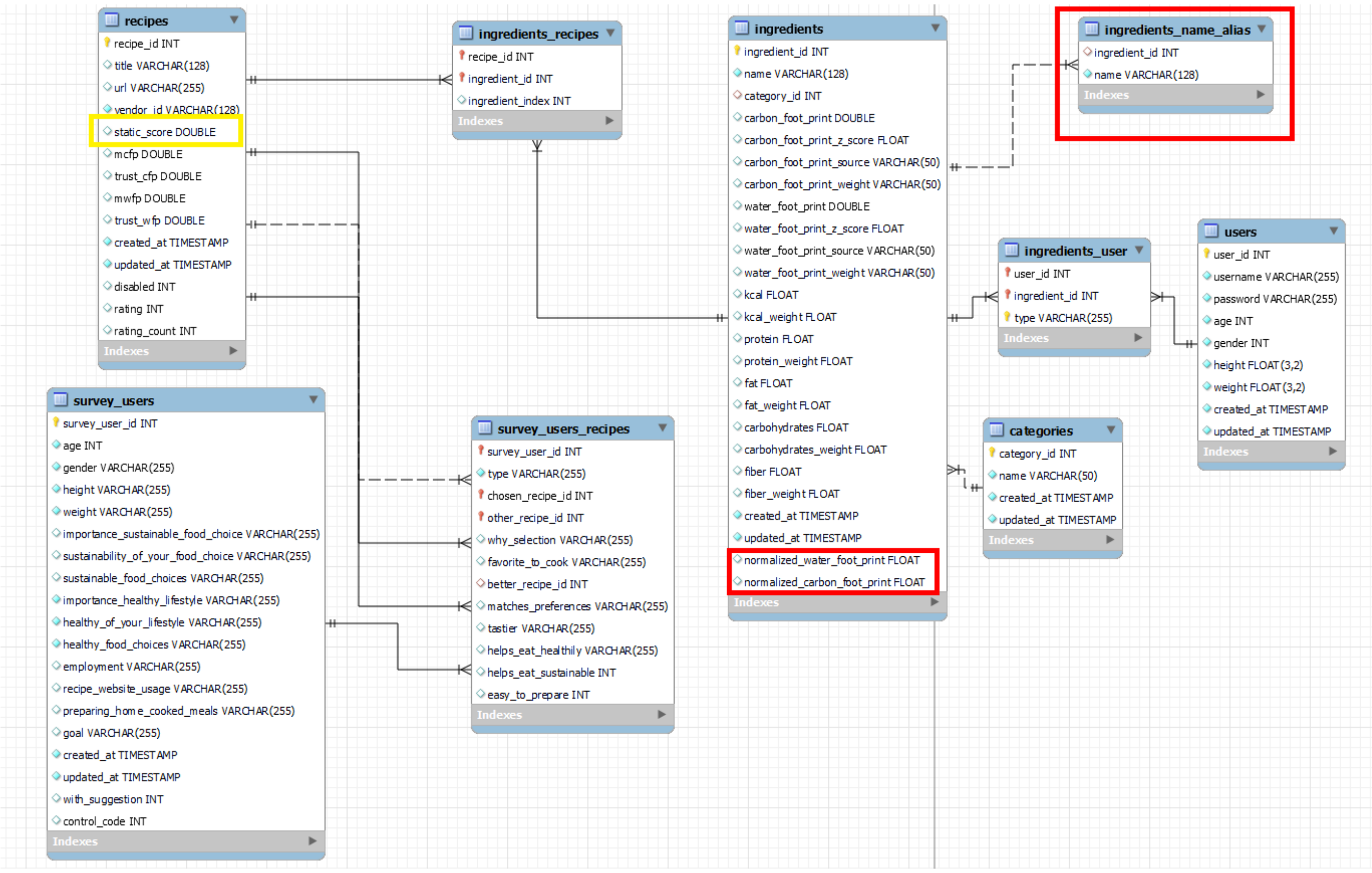
There are also matches that were derived entirely manually, this derives from some corrective action upon the automatic matching algorithm that was integrated directly into the final set of updates without checking if there was an automatic way to derive the same information.

It's important to say that there still are ingredients in FoodPrintDB without any information about cfp and wfp, this is because in the SEL database and so in the derived CSEL dataset there wasn't any proper food item or food typology that could be relatable to these ingredients. Fortunately, we are talking about 70 ingredients over 352, and most of them are ingredients not so important in the recipes.

An analysis regardings how to expand the SEL database in order to also include the proper food item and/or food typologies for the missing ingredients would be a good starting point for future works on this topic.

FoodPrintDB_v2 structure

The final database has the following structure[1]:



Where in red there are the new elements not already present in FoodPrintDB_v1 and in yellow are the elements (just one indeed) which have their semantics changed in the new version.

The table ingredient is so defined (in red the new fields):

Field Name	Type	Description
ingredient_id	int	Unique identifier for each ingredient.
name	varchar(128)	The name of the ingredient.
category_id	int	The category ID this ingredient belongs to.
carbon_foot_print	double	The carbon footprint of the ingredient.
carbon_foot_print_z_score	float	The carbon footprint z score of the ingredient. (not used in the new version)
carbon_foot_print_source	varchar(50)	The source from where the carbon footprint value has been retrieved.
carbon_foot_print_weight	varchar(50)	The weight of the ingredient used to compute the cfp.
water_foot_print	double	The water footprint of this ingredient.

[1] The tables users, ingredients_users, survey_users and survey_users_recipe are tables strictly related to specific functionalities of the FoodPrint web app developed by the Amoruso and Fusillo, will not be described in this section. Refer to the original documentation for further information.

water_foot_print_z_score	float	The water footprint z score of the ingredient. (not used in the new version)
water_foot_print_source	varchar(50)	The source from where the water footprint value has been retrieved.
water_foot_print_weight	varchar(50)	The weight of the ingredient used to compute the wfp.
kcal	float	The energy content of this ingredient in kilocalories.
kcal_weight	float	The weight of the ingredient used to compute the kcal.
protein	float	The protein content of this ingredient.
protein_weight	float	The weight of the ingredient used to compute the amount of protein.
fat	float	The fat content of this ingredient.
fat_weight	float	The weight of the ingredient used to compute the amount of fat.
carbohydrates	float	The carbohydrate content of this ingredient.
carbohydrates_weight	float	The weight of the ingredient used to compute the amount of carbohydrates.
fiber	float	The fiber content of this ingredient.
fiber_weight	float	The weight of the ingredient used to compute the amount of fiber.
created_at	timestamp	The timestamp when this ingredient was created.
updated_at	timestamp	The timestamp when this ingredient was last updated.
normalized_water_foot_print	float	The normalized water footprint of this ingredient.
normalized_carbon_foot_print	float	The normalized carbon footprint of this ingredient.

The category table is so defined:

Field Name	Type	Description
category_id	int	Unique identifier for each category.
name	varchar	Name of the category.
created_at	timestamp	The date and time when the category was created.
updated_at	timestamp	The date and time when the category was last updated.

The ingredients_name_alias table is a new table present in FoodPrintDB_v2 with the aim of containing an alternative name for the ingredient, these new names could be easier to use in an nlp context. The table is so defined:

Field Name	Type	Description
ingredient_id	int	Reference to the aliased ingredient id.
name	varchar(128)	Alias name for the referenced ingredient.

The recipes table is so defined (in yellow the field whose semantic is changed):

Field Name	Type	Comment
recipe_id	int	Unique identifier for each recipe.
title	varchar(128)	Title of the recipe.
url	varchar(255)	URL of the recipe.
vendor_id	varchar(128)	Identifier for the vendor associated with the recipe (not used).
static_score	double	Static score for the recipe, used to rank the recipe in search results.
mcfp	double	Average cfp of the recipe, not used in FoodPrintDB_v2.
trust_cfp	double	Percentage of related ingredients which have a cfp.
mwfp	double	Average wfp of the recipe, not used in FoodPrintDB_v2.
trust_wfp	double	Percentage of related ingredients which have a wfp.
created_at	timestamp	Timestamp for when the recipe was created.
updated_at	timestamp	Timestamp for when the recipe was last updated.
disabled	int	Flag for disable a recipe in the webapp.
rating	int	Average rating of the recipe.
rating_count	int	Number of rates given to the recipe.

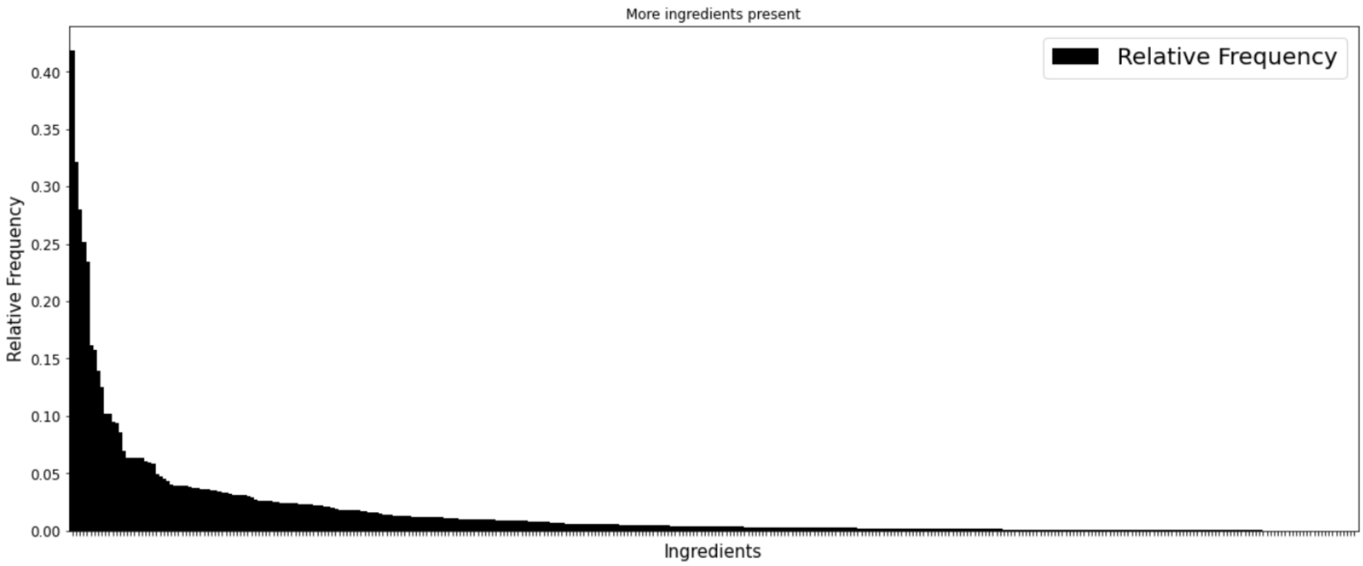
The new formula used to compute the static score is explained in the “Sustainability score computation” section later on.

The ingredients_recipe table allows one to relate each recipe to their ingredients and is so defined:

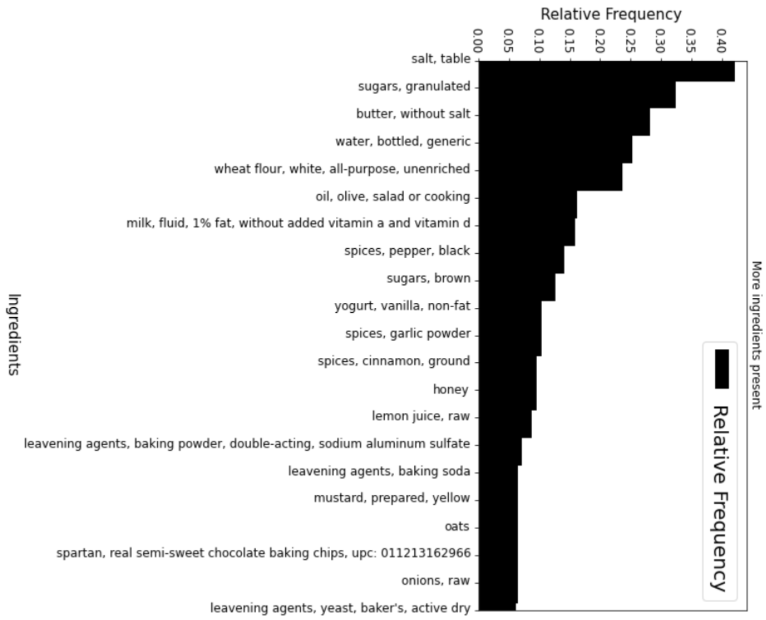
Field Name	Type	Description
ingredient_id	int	Reference to the id of the ingredient involved in the recipe.
ingredient_id	int	Index of the ingredient which allows ordering the ingredient in the recipe.
recipe_id	id	Reference to the id of the recipe.

Comparison: FoodPrintDB_v1 vs FoodPrintDB_v2

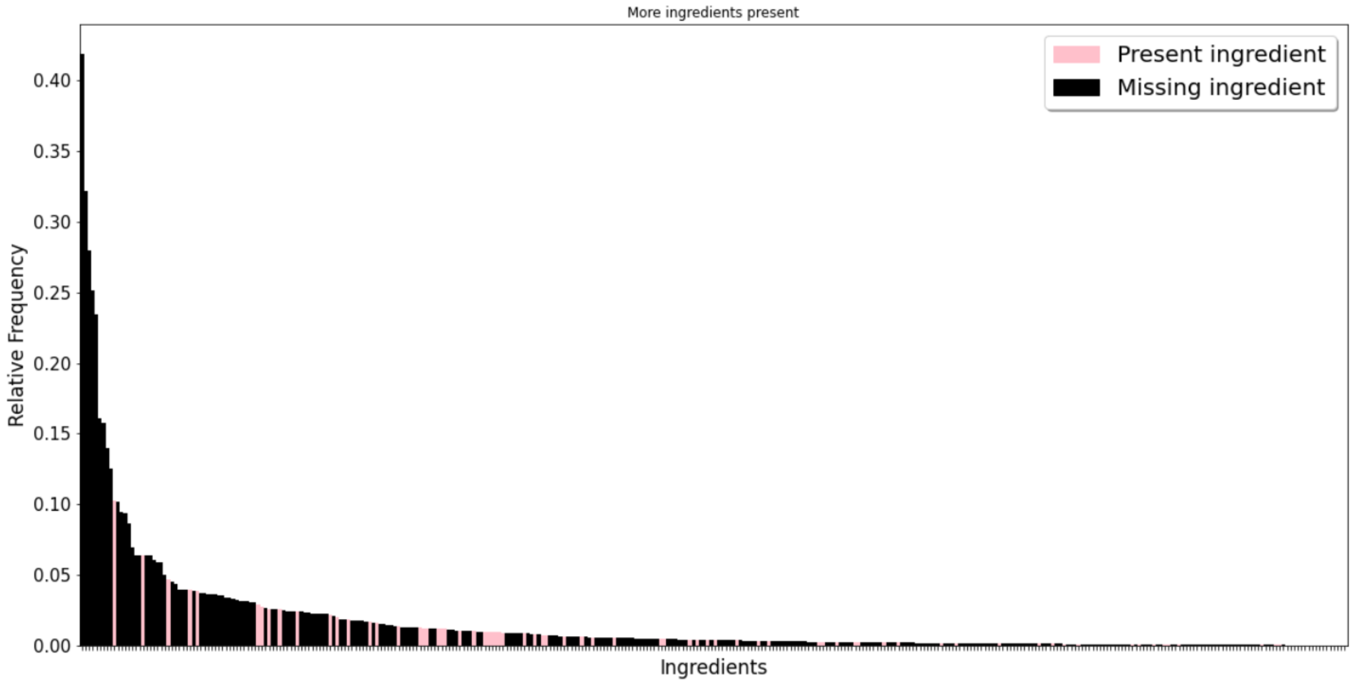
Using the previously described techniques we were able to increase the number of ingredients that own both the information about cfp and wfp from a value of 72 for FoodPrintDB_v1 to a value of 292 for FoodPrintDB_v2, in this work such ingredients are called “valid ingredients”.
A more interesting comparison could be done by plotting the relative frequency of each ingredient with respect to the recipes in which it appears and sorting the results in a descending manner.



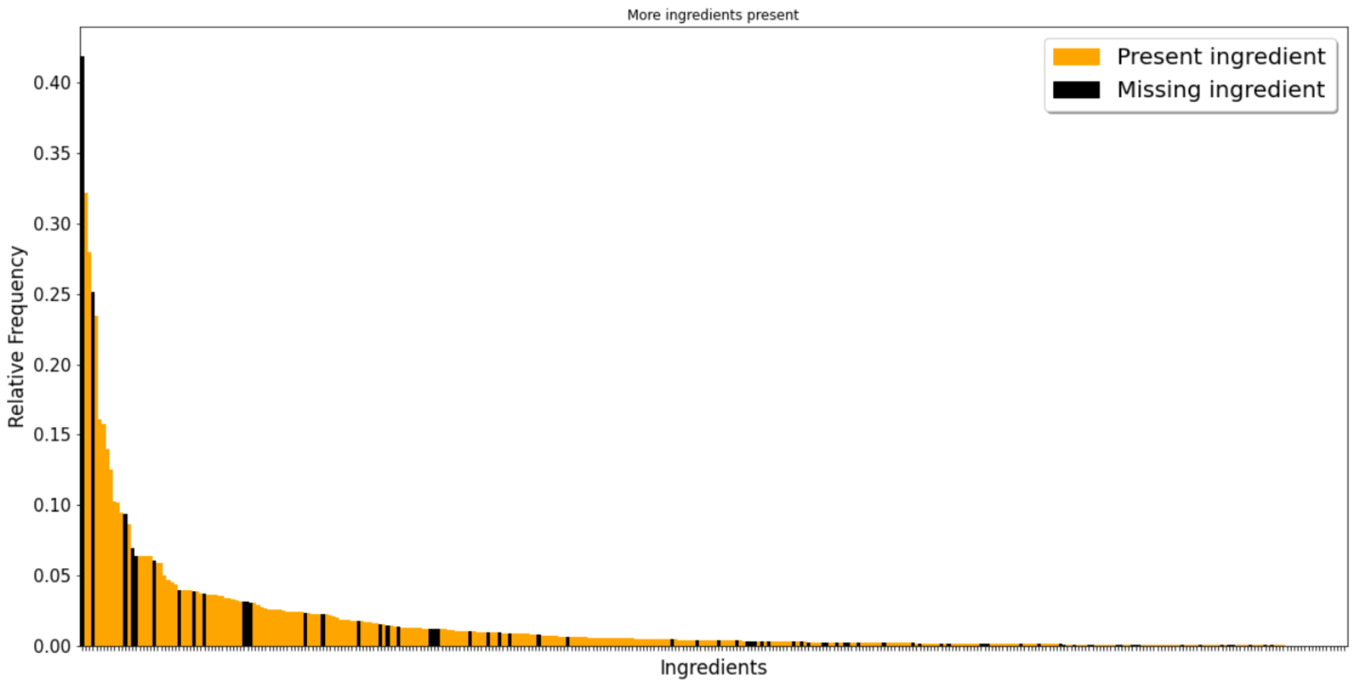
The shape of this curve resembles evidently the so-called Zipf Curve, which means that there is a small subset of ingredients that are really frequent in most of the recipes and a large subset of less important ingredients. In order to have a decent dataset of recipes whose ingredients are valid, at least the elements in the head and in the body should be valid.



Here we have a zoom on the head and the body of the curve, these are the most important ingredients in the database.



In this chart are plotted in pink the ingredients that are valid in the FoodPrintDB_v1 database, it's possible to see that the most important ingredients are not valid, this explains why the recipes where at least 80% of ingredients are valid are just 357.



In this chart are plotted in orange the ingredients that are valid in the FoodPrintDB_v2 database, it's possible to see that most of the ingredients are valid, especially the ones in the head and in the body. This is an important result since it conveys an important increase in the number of recipes composed of valid ingredients.

Recipe coverage in FoodPrintDB_v1

Recipe coverage is calculated using a threshold that considers the number of valid ingredients in each recipe. The table shows the recipe coverage for three different thresholds: 80%, 90%, and 100%.

As already stated, valid ingredients are those for which we were able to associate a value of cfp and wfp with them. Only recipes for which the number of valid ingredients is at least as the specified threshold (80%, 90%, or 100%) are considered valid.

For example, if a threshold of 80% is considered, a recipe will be considered valid if at least 80% of the ingredients present in it are valid ingredients. This means that for a recipe consisting of 10 ingredients, at least 8 must be valid ingredients for the recipe to be considered valid.

In this way, it is possible to calculate the number of valid recipes for each threshold and therefore the overall coverage of the recipes available for analysis. For the sake of completeness, these analyses are also performed taking into account just the cfp or just the wfp.

Total number of recipes: 51 235	**foodprintdb_v1**		
	cfp	wfp	both
80%	1 360	357	357
90%	905	295	295
100%	905	295	295

The table shows that even considering a coverage of the 80% there are only 357 with both the information about cfp and wfp. A better result is available only if we limit our concept of validity to those ingredients with just the cfp, obtaining a number of 1360 recipes, that is however a really small number compared with the totality of recipes. These really low numbers are easily explainable considering that the most frequent ingredients are not covered by this version of the database.

Recipe coverage in FoodPrintDB_v2

The integration of CSEL data in the FoodPrintDB has made it possible to obtain a wider coverage of recipes available for analysis. For example, if we consider an 80% threshold, the number of valid recipes is 23503, which means that almost half of the initial recipes were considered valid and ensures a big number of recipes available for analysis. Further increasing the threshold to 90%, the number of valid recipes drops to 12342, but this is still a significant number of valid recipes. Finally, even with a threshold of 100%, the number of valid recipes is 11492, which represents a decent number of recipes available for analysis.

Total number of recipes: 51.235	footprintdb_v2		
	cfp	wfp	both
80%	34 662	23 503	23 503
90%	19 371	12 342	12 342
100%	17 691	11 492	11 492

The wide coverage reached by this new version of the database provides more recipes on which it is possible to test different methods for computing a score which represents the overall sustainability of the recipes in terms of carbon footprint and water footprint.

Sustainability score computation (static score)

The sustainability score is used to evaluate the environmental sustainability of the recipes, i.e. how much the recipes affect the environment in terms of carbon footprint and water footprint. The score helps identify recipes that have a lower environmental impact, i.e. those that are more sustainable. This can be useful for people who want to make more sustainable choices in their diet or for researchers who want to study the environmental impact of recipes. This information is stored into the "static_score" field of the recipes table.

The old formula

Originally the static score was computed through this formula:

$$Ss(r) = \frac{Zss(r) - MinZss}{MaxZss - MinZss}$$

$$Zss(r) = \sum_{j=1}^{|J|} Zcfp(J_i) + \sum_{i=1}^{|J|} Zwfp(J_i)$$

where *Zss(r)* is the sum of z-scores of the cfp and wfp related to the ingredients involved in the recipe, while MaxZss and MinZss are the maximum and the minimum value of Zss among all the recipes and are used to apply a min-max normalization on the score.

This formula was based on the concept of having a neutral value by means of the z-normalization and bringing both cfp and wfp on the same magnitude. Unfortunately, once this formula has been applied considering the recipes in FoodPrintDB_v2, we discovered that the z-normalization, since preserving the ratio of the original values, made some really high wfp values so important to become crucial in the recipe score computation. This phenomenon made some recipes that own highly demanding water spices like saffron the most unsustainable recipes in the database. This wasn't good since we know that spices are used in small quantities and wouldn't impact sustainability so much. We also know that in the most consumed ingredients like meat and

animal-derived products, the most dangerous aspect is the carbon footprint, so we decided to find a way to make it more relevant in the score computation.
The first try was to simply scale down the wfp using two new factors α and β where α is related to cfp and β to wfp modifying the formula of the Zss(r) as follow:

$$Zss(r) \;=\; \alpha \sum_{i=1}^{|J|} Zcfp(J_i) \;+\; \beta \sum_{i=1}^{|J|} Zwfp(J_i)$$

Comparing the recipe scoring using two different set of values for α and β we obtain the following results:

α=0.5, β=0.5	
title	static_score
Pain d'Epices	1
Ras el Hanout	0,976007616
Sikarni (Spiced Sweet Yogurt-Pistachio Dessert)	0,9737026478
Shrikhand and Pooris	0,9603371213
Sholeh Zard- Saffron Rice Pudding	0,9578134048
Saffron and Carrot Halva	0,9411938181
Sunshine Bitters Recipe	0,9406670102
Rose Ice Cream	0,9395126287
Flavored Butter: Saffron and Cardamom Butter	0,9393501429
Kau Katli or Cashew Fudge (Vegan)	0,9389434757

static score α=0.999, β=0.001	
title	static_score
Pain d'Epices	1
Sikarni (Spiced Sweet Yogurt-Pistachio Dessert)	0,970997046
Ras el Hanout	0,9643037133
Shrikhand and Pooris	0,9587892081
Sholeh Zard- Saffron Rice Pudding	0,9531892648
Flavored Butter: Saffron and Cardamom Butter	0,9475174806
Sunshine Bitters Recipe	0,9412748582
Rose Ice Cream	0,9410555912
Kau Katli or Cashew Fudge (Vegan)	0,9372326062
Sweet Vermicelli Breakfast	0,9358204205

Is possible to see that comparing the basic scenario where α and β are both 0.5 (same importance, equivalent to the original formula) to the one were α=0.999, β=0.001 the lists of most unsustainable recipes don't change so much. The recipe "Pan d'Epices", a kind of bread rich in spices, is still considered the most dangerous recipe, even more than recipes that are rich in meat and animals derived products that even doesn't appear on the lists. This was due to the really big difference in terms of the ratio between cfp and wfp and suggest using an alternative approach.

To summarize, despite the introduction of alpha and beta, the results obtained were poor. The main reason for this failure is the fact that the wfp continues to have a significant impact on the final outcome.

First solution

For this reason, we decided to introduce a new formula that uses both the previously normalized cfp and wfp. In this way, both values move in the same range and have the same ratio, allowing us to obtain more meaningful values.
The new formula is the following one:

$$Ss(r) = \frac{Nss(r) - MinNss}{MaxNss - MinNss}$$

$$Nss(r) = \alpha \sum_{i=1}^{|J|} Ncfp(J_i) + \beta \sum_{i=1}^{|J|} Nwfp(J_i)$$

$$MinNss = min_{\forall r \in R} Nss(r)$$

$$MaxNss = max_{\forall r \in R} Nss(r)$$

$$Ncfp(x) = \frac{cfp(x) - minCfp}{maxCfp - minCfp}$$

$$Nwfp(x) = \frac{wfp(x) - minWfp}{maxWfp - minWfp}$$

$$minCfp = min_{\forall x \in X} cfp(x)$$

$$minWfp = min_{\forall x \in X} wfp(x)$$

$$maxCfp = max_{\forall x \in X} cfp(x)$$

$$maxWfp = max_{\forall x \in X} wfp(x)$$

Where *J* is the set of valid ingredients related to a recipe, *R* is the set of all the recipes, *X* is the set of all the ingredients in the dataset, and α, β are two factors used in order to change the impact of cfp and wfp if desired.
This new formula, using α=0.5, β=0.5, allows us to obtain more coherent results with the idea that recipes that contain meat and animal-derived products must have a higher score as we can see in the following table:

title	static_score
Sour Cream Noodle Delight	1
Herb Pie Recipe	0.9514996607467779
Barbecue Bacon Cheeseburger Turnovers	0.9226651186788645
Eight-Layer Casserole	0.9068370981670131
Quinoa Risotto with Broccoli and Saffron	0.9006579464748306
Brown Rice With Cashews; Spinach; And Mushrooms Recipe	0.8846858127272832

Cheeseburger Soup	0.8744895371455301
Rip Roaring Runzas (aka Cornhusker Calzones)	0.8739034261685281
Mushroom Beef Patties	0.8595663708682267
Cheeseburger Soup	0.8577832224547433

However, further analyses on the results shows that recipes composed of one single, medium sustainable, ingredient are more sustainable than a recipe that is composed of several highly sustainable ingredients. We can notice this phenomenon by comparing a recipe that contains only pork with a salad. We can notice this issue by comparing a recipe that contains only pork with a salad.

title	static_score
Roast Pork Shoulder	0.061877226633926814
Salad of Cold Leafy Greens With Sesame Oil and Soy Sauce	0.08099551767122119

This phenomenon was due to the fact that the formula adds more and more cfp and wfp value without taking into account their real quantities in the recipe, so it is calculated as if there were 1 kg for each ingredient involved.

Second solution: importance-based sustainability score

To prevent this behavior, we have made significant changes to the scoring function. We now follow the principle that, without knowledge of the exact quantities of each ingredient, the only reliable criterion is to score the recipe list in the same order as the ingredient list taking into account the main ingredient of the recipe. The main ingredient, with the highest sustainability score, is prioritized, and additional ingredients are discounted based on their importance. The less important the ingredient, the stronger the discount, where the importance is how much an ingredient negatively contributes from the sustainability point of view.
In this way, the score of each recipe will be based on the score of the main ingredient and also has the possibility to give more weight to recipes that share the same main ingredient but have more additional ingredients.
It is important to point out that in this new formula, the sustainability of an ingredient is computed a priori as a linear combination of normalized cfp and wfp using two factors α and β adjusted following the idea that meat and ingredients of animal origin should have more importance.
The new formula is so defined:

$$Ss(R) = \frac{Dss(R) - MinDss}{MaxDss - MinDss}$$

$$Dss(r) = \sum_{i=0}^{|K|-1} Iss(K_i)\, e^{-i}$$

$$Iss(x) = \alpha\, Ncfp(x) + \beta\, Nwfp(x)$$

Where K is the set of ingredients related to a recipe ordered on the base of their sustainability score in descending order, so K_0 will be the most unsustainable ingredient of the recipe and $K_{|K|-1}$ will be the most sustainable ingredient in the recipe.
The normalized cfp and the normalized wfp are computed in the same way of the previous formula, so they are not reported again.
The two factors, α, and β, are respectively set to 0.8 and 0.2 in order to give more importance to cfp rather than wfp and allow animal-derived products (primarily beef meat and cheese) to be ranked as the most unsustainable ingredients.
It's possible to see how this new formula scores the most unsustainable recipes placing at the top recipes which contain beef meat, for example, the 10 most unsustainable recipes now are:

title	static_score
Marinated BBQ Flank Steak	1
Ribeye With Fresh Tomato Salad	0.9183409077443927
Beef with Romesco Sauce	0.867106861355729
Arrachera Beef Marinade	0.8504107429527841
Fajita Marinade II	0.8487424191633927
Tahitian Grilled Steak	0.8476198979631431
Shaker Flank Steak	0.8462726935199699
Pot-Browned Noodles With Beef and Leeks	0.845848543719871
Grilled Peppery London Broil	0.845705819378264
Italian Flank Steak	0.8456717451561896

Furthermore, as we can see, the previously shown problem with the example of the pork recipe and the salad is no more present.

title	static_score
Roast Pork Shoulder	0.10103848523439829
Salad of Cold Leafy Greens With Sesame Oil and Soy Sauce	0.07779811463596747

This new scoring formula follows common sense without necessarily requiring information on the exact amount of each ingredient and can easily be used to create a number of different recipes based on their sustainability score.

Transformers applications as suggestions for dataset use

Transformer models are the state of the art in today's natural language processing applications, and several libraries are available that allow developers to build powerful applications on complex pre-trained models. To explore the potential offered by the data in FoodPrintDB_v2 let's try to build some small demos that solve some tasks related to food.
In particular, we try to build:

- A recipe classifier able to classify a recipe among three different sustainability classes.
- A recipe recommender who, by providing a list of ingredients, is able to suggest some recipes at different levels of sustainability, giving this suggestion by generating a non-repetitive text in natural language.

To proceed with the analysis of these two works, it is first of all necessary to understand how the data present in FoodPrintDB_v2 were managed in order to construct a dataset of recipes.

Preprocessing

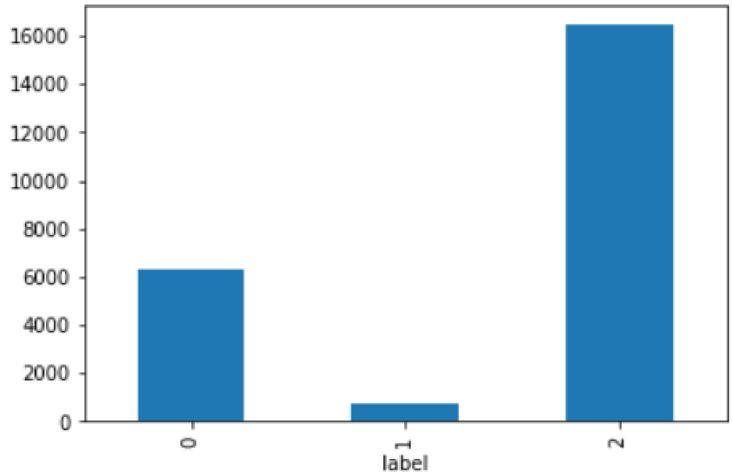
Choose recipes

We have chosen the recipes that contain a significant percentage of sustainable ingredients both in terms of carbon and water, filtering out those with a "recipe_coverage" greater than or equal to 80%. This threshold represents a limit that has been chosen to represent a "good" coverage of sustainable ingredients, so as not to include recipes that contain only a few sustainable ingredients or only a small amount of sustainable ingredients.

Categorization

The recipes are ranked according to their sustainability, using the "static_score" as an indicator. Recipes are labeled as "LOW" (score >= 0.5), "MEDIUM" (score between 0.1 and 0.5), or "HIGH" (score <= 0.1) sustainability. This subdivision allows you to easily select a recipe based on the sustainability of the ingredients it contains.

Below is the graphic count of the labels (where HIGH=0, LOW=1, MEDIUM=2):



At the end of the preprocessing, a subset of the dataset useful for training was selected, the structure is described as follows:

recipe_id	title	url	ingredients	sustainability
3049	Create Your Own Bars Recipe	http://cookeatshare.com/recipes/create-your-ow...	wheat plain flour,butter,cane sugar	medium
23723	Maui Mustard Sauce	http://www.food.com/recipe/maui-mustard-sauce-...	apricot,pineapple,yellow mustard	high
46208	Chunky Chicken Nachos	http://www.food.com/recipe/chunky-chicken-nach...	ceddar,olives,salsa,chicken bone free meat	medium
25296	V's Chile Verde	http://www.food.com/recipe/vs-chile-verde-268423	onion powder,red pepper or cayenne spices,onio...	low
49261	Microwave Spicy Pepper Steak Recipe	http://cookeatshare.com/recipes/microwave-spic...	beef meat with bone	low
24585	Savory Steak Rub	http://www.food.com/recipe/savory-steak-rub-19...	dried rosemary,red pepper or cayenne spices,ga...	high
…	…	…	…	…

Following the instructions on the hugging face documentation, we extract the useful columns in the required format, so before training the dataset will look like this:

recipe_id	label	text
24585	high	dried rosemary,red pepper or cayenne spices,ga...
23723	high	apricot,pineapple,yellow mustard
25296	low	onion powder,red pepper or cayenne spices,onio...
49261	low	beef meat with bone
3049	medium	wheat plain flour,butter,cane sugar
46208	medium	ceddar,olives,salsa,chicken bone free meat
…	…	…

Classification with roberta-base

Training

A sequence classification model is trained[9] using the model "**roberta-base**"[10] on a dataset of 600 examples, divided into a training set and a test set with a percentage of 80% and 20%, respectively, and reaches an accuracy of about 96.4%.

We conducted a study to evaluate the sustainability of different dishes based on their ingredients by making predictions for each class.

For the first example, the ingredients "ginger, tofu, soy sauce, carrot": The sustainability of this ingredient combination is high because tofu and soy sauce are often produced sustainably and ginger root and carrots can be grown sustainably.

In contrast, the second example, which included "onion, paprika, fresh parsley, black pepper spice, celery, bone-in beef, brown sugar": The sustainability of this ingredient combination is low because beef is generally considered unsustainable due to the impact on the environment of meat production and welfare. Additionally, brown sugar can be produced unsustainably. The other ingredients, such as onion, paprika, parsley, black pepper, and celery, can be grown sustainably.

Finally, we reviewed a dish containing "Strawberry semi-sweet chocolate chips": The sustainability of this ingredient combination is medium, maybe because strawberries can be grown relatively sustainably, while semi-sweet chocolate and chocolate chips may contain few sustainable ingredients, such as cocoa butter or refined sugar.

Our results suggest that careful consideration of ingredients is essential for determining the sustainability of dishes.

Food recommendation using transformer models

The goal of our recommender system is to provide users suggestions about sustainable recipes in natural language based on a list of ingredients provided by them. Basically, the user enters the ingredients that he/she has available or that he/she wants to use to prepare a dish, and the system suggests the most suitable recipes. The ultimate goal is to provide the user with a selection of sustainable recipes that take into account his/hers food preferences helping him/her to choose what to cook quickly and easily.

Calculate similar recipes using all-MiniLM-L6-v2 embeddings

The algorithm calculates the similarity between the ingredients present in the input and those present in a selection of recipes that have high and medium degrees of sustainability to find recipes more similar to the one requested by the user. In particular, two different subsets of the dataset are selected, the ones which have the string 'HIGH' or 'MEDIUM' as the value of the 'SUSTAINABILITY' column, then the ingredients of interest are encoded using the embedding model all-MiniLM-L6-v2[11]. Subsequently, the cosine similarity between the encoded ingredients of interest and those encoded via the embedding model is calculated, then a subset of three recipes for each of the two groups with the highest similarity is selected.

For instance, given the list of ingredients "oats,banana,milk", the similar recipes are calculated by computing the similarity between the embedding of our input and the selected recipes (the subset with high sustainability in this case/example) using the "all-MiniLM-L6-v2" model with SentenceTransformer. The three best results for each sustainability level are shown in the following tables.

index	recipe_id	title	url	ingredients	SUSTAINABILITY
487	35477	Survival Necklace	http://www.food.com/recipe/survival-necklace-95937	banana,oat meal	HIGH

1951	50516	Strawberry Banana Smoothie	http://www.kraftrecipes.com/recipes/strawberry-banana-smoothie-88100.aspx	cow milk,banana,strawberry,oat meal	HIGH
1728	2818	Oatmeal and Bananas Breakfast (Baby Food)	http://www.food.com/recipe/oatmeal-and-bananas-breakfast-baby-food-52539	oat,cow milk,banana	HIGH

index	recipe_id	title	url	ingredients	SUSTAINABILITY
5837	23711	Oatmeal With Blueberries and Cream	http://www.food.com/recipe/oatmeal-with-bluebe...	oat,cream,cow milk,blueberry	MEDIUM
8495	39484	Apple Butter Oatmeal	http://www.food.com/recipe/apple-butter-oatmea...	oat,cow milk,butter	MEDIUM
1461	45835	Fruit and Malt Oat Shake	http://www.food.com/recipe/fruit-and-malt-oat-...	oat,cow milk,banana,semi-sweet chocolate bakin...	MEDIUM

From these two recipe sets, only one recipe per set is selected; the one that is closer to the ingredient list provided as input. We have chosen to consider two different recipes at different levels of sustainability to allow the users to select the one that is more recognizable with their taste and also to avoid suggesting a strictly vegetarian diet since all the recipes in the "HIGH" category sustainability sets are basically vegetarian recipes.

Then the following methods will use the two results obtained from the similarity calculation to produce a suggestion in natural language.

Generating the answer with EluetherAI/gpt-neo-2.7 B and GPT-3/text-davinci-003

GPT-Neo and GPT-3 are high-performance natural language models developed by EluetherAI and OpenAI, respectively. GPT-Neo was released as an open-source alternative to GPT-3, as the latter was only released for commercial use. Both models use machine learning to generate text in a coherent and natural way, using a huge amount of data to learn language patterns[12]. In this context, we will use the EluetherAI/gpt-neo-2.7 B model to generate the recommender system's responses in order to provide a smooth and natural response to the users[13].

In both cases, there is a prompt given in input to the model capable of generating the response, although in the case of text-davinci-003, only one prompt was sufficient, in the case of GPT-Neo the generation of the text was guided. Both examples accept a prompt and return the recommendation.
In the prompts that will be described in the following section, we use some placeholder in angular brackets to indicate pieces of strings that are variables dependent:

- `<ingredients_list>` is replaced with the ingredients list provided in input.
- `<mid_sus_recipe_name>` is replaced with the name of the medium sustainable recipe.
- `<mid_sus_ingredients_list>` is replaced with the ingredients list of the medium

sustainable recipe.

- <mid_sus_recipe_url> is replaced with the url of the medium sustainable recipe where it is possible to read its procedure.
- <sus_recipe_name> is replaced with the name of the sustainable recipe.
- <sus_ingredients_list> is replaced with the ingredients list of the sustainable recipe.
- <sus_recipe_url> is replaced with the url of the sustainable recipe where it is possible to read its procedure.

1. EluetherAI/gpt-neo-2.7B

The GPT-Neo 2.7B model is a transformer model, based on the OpenAI GPT-3 architecture replicated by EleutherAI, and has been trained on a dataset called "Pile" to generate text from a prompt, but can also be subjected to fine-tuning for specific tasks such as classifying text or answering questions. The model outperforms GPT-3 Ada on several linguistic reasoning tasks, such as natural language understanding and understanding context, but GPT-3 Ada outperforms some scientific reasoning tasks, such as answering medical questions and understanding images.

We simply call the tokenizer, instantiate the model, initialize the pipeline for generation, and give the prompt, then concatenating the different generated text (each one for each attribute in the resulting recommendation table) we can obtain a good answer.

For each level of sustainability (high and medium therefore), we composed prompts that guide the generation of the suggestion once combined with the suggestions actually obtained from the similarity calculation. The two prompts are so defined:

```
prompt1 = “A recipe <ingredients_list>”

prompt2 = “What do you think about <mid_sus_recipe_name|sus_recipe_name>
it has <high|medium> sustainability”
```

And when they are used to generate the two suggestions for each different sustainability level the results are of these kinds:

Medium sustainability suggestion:

> prompt1: “A recipe cow milk,banana,strawberry,oat meal is”
>
> generated text1:
>
> “A recipe oat,cow milk,banana,semi-sweet chocolate baking chipsThe first thing I do every morning is run to the refrigerator to check for the banana. I like”
>
> prompt2:
> “What do you think about Fruit and Malt Oat Shake? it has medium sustainability”

generated text2:

“What do you think about Fruit and Malt Oat Shake? it has medium sustainability score.\n\nI am trying to buy it a long time before a long time and I don't have”

High sustainability suggestion

prompt1: “A recipe oat,cow milk,banana”

generated text1:

“A recipe oat,cow milk,banana cream and pineapple juice with sweetened shredded coconut and fresh pineapple,tossed with a little grated ginger in a bowl made into a mold and”

prompt2: “What do you think about Strawberry Banana Smoothie? it has high sustainability.”

generated text2:

“What do you think about Oatmeal and Bananas Breakfast (Baby Food) it has high sustainability.Do you like the idea of Organic Bananas, Oatmeal (Breakfast Cereal)”

So by composing more generated text we can get a response from the system that is not repetitive, in this case by adding a string with the link:

1. A recipe oat,cow milk,banana,semi-sweet chocolate baking chipsThe first thing I do every morning is run to the refrigerator to check for the banana. I like.What do you think about Fruit and Malt Oat Shake? it has medium sustainability score.I am trying to buy it a long time before a long time and I don't have, you can find more on: http://www.food.com/recipe/fruit-and-malt-oat-shake-463414
2. A recipe oat,cow milk,banana cream and pineapple juice with sweetened shredded coconut and fresh pineapple,tossed with a little grated ginger in a bowl made into a mold and.What do you think about Oatmeal and Bananas Breakfast (Baby Food) it has high sustainability.Do you like the idea of Organic Bananas, Oatmeal (Breakfast Cereal), you can find more on: http://www.food.com/recipe/oatmeal-and-bananas-breakfast-baby-food-52539

The reason we give this type of prompt is that unlike GPT-3 (as we will see in the next example) it is not possible to make a direct request to the model because it has been "guided" in generating the text. The so-generated suggestions tend to be nonsense, we believe the model should be trained with an appropriate example of a recommendation to enable it to generate consistent text in this context, or, as an alternative, this model could be used to generate only a small portion of a text that must be used in a more structured template.[2]

2. GPT-3/text-davinci-003

[2] Warning: EluetherAI/gpt-neo-2.7B is capable of generating offensive content.

To overcome the limitation of the previous model we also try to generate a recommendation using the GPT3 model "text-davinci-003".

To allow the model to generate a suggestion with the necessary information, the prompt includes information about the most sustainable recipes in the "HIGH" and "MEDIUM" datasets, as well as their ingredients and URLs.

The prompt also tells the model how to suggest recipes based on the ingredients selected by the user and how to order the two different suggestions. We have chosen to suggest the average sustainable recovery as the first option and then suggest the more sustainable recipe as an alternative if the user wants to maximize the sustainability of their food choices.

The prompt used is so defined:

```
"I'm building a recommender system for recipes. The user has selected the
following ingredients: <ingredients_list>.\nsuppose we have a medium
sustainable recipe <mid_sus_recipe_name> composed of
<mid_sus_ingredients_list> which is readable at url <mid_sus_recipe_url> and
a fully sustainable recipe <sus_recipe_name> composed of
<sus_ingredients_list> which is readable at url  <sus_recipe_url> . \nWrite a
text that suggests, referring to the user's ingredients, to consume the first
recipe and the second if he/she prefers a more sustainable recipe.\nYou must
point out that the second recipe is more sustainable.\nBoth suggestions must
refer to the url in order to make possible for users to read the full
recipes.\nIf you refer to recipes ingredients use only the ones listed."
```

So, for example, given an input like “oats,banana,milk” the placeholders in the prompt are replaced with the computed similar recipes information, and then the prompt is given in input to the model. As result, the following text is produced:

> If you’re looking to make a delicious, nutritious breakfast with your ingredients of oats, banana, and milk, you should try this delicious Fruit and Malt Oat Shake! This recipe is composed of oats, cow milk, banana, and semi-sweet chocolate baking chips and can be found at http://www.food.com/recipe/fruit-and-malt-oat-shake-463414. For a more sustainable recipe, try out this Oatmeal and Bananas Breakfast (Baby Food) recipe composed of oats, cow milk, and banana at http://www.food.com/recipe/oatmeal-and-bananas-breakfast-baby-food-52539. It is a more sustainable option compared to the Fruit and Malt Oat Shake.

It’s possible to see that the generated suggestion takes into account all the conditions expressed in the prompt allowing one to obtain a coherent suggestion with all the information that the user needs to select a proper recipe. A strength of this model is that it is capable of generating text taking into account several constraints and conditions without the need for further training processes, moreover, the generated text is expressive and non-trivial, so it can be easily used in a chatbot system that can be integrated into the everyday life use. One limitation of GPT3 is that the

API call to the system is not for free[3], so this kind of solution could not be an economically-sustainable choice for non-commercial applications.

Conclusions and future works

The work carried out on the dataset certainly represents a significant step forward both in terms of quantity and quality of the data and is a valid starting point for sustainability-related tasks as viewed in the experiments done in the last section. However, there is still room for future improvements, for example, it is possible to try to integrate those missing ingredients which, despite having a relatively high frequency, are not yet present in the dataset. This could be done by performing some searches in white and grey papers about the emission and the water consumption of the ingredient still not covered and including them in the database following the methods already provided by the SuEatable Database authors. Furthermore, some machine learning techniques could be also used to infer cfp and wfp for missing data, however, it is necessary to understand clearly which data could play the role of features in this context.
The database FoodPrintDB is also improvable by increasing the number of ingredients and recipes available, in order to increase the number of ingredients there are several food items in the SEL database that could be easily integrated into a new version of the database.
A further, crucial, improvement would be played by expanding the ingredients_recipe table with information about the quantity of each ingredient. This information would help to compute a recipe sustainability score capable of reflecting the real impact of each recipe both on the emission side and the water consumption side. Other interesting information that could complete the database is the one related to the kind of cooking methods and the time needed to prepare a dish. These aspects play an important role in terms of co2 emissions and must be taken into account when choosing a recipe.
As the last point that must be considered for future works, there is the kind of algorithms used in order to manipulate the available information. Several models, like the already viewed GPT-3, run on heavily demanding servers that contribute themselves to the problem that we are trying to solve, so is also necessary to understand if more sustainable choices can be done also from the computational point of view preferring, ideally, preferring simpler models or even more traditional approach like rule-based AI at cost of having less impressive results[14].

Bibliography

1. EU Climate and Energy Framework for 2030: https://climate.ec.europa.eu/eu-action/climate-strategies-targets/2030-climate-energy-framework_it
2. Food Recommender Systems: Important Contributions, Challenges and Future Research Directions by Christoph Trattner and David Elsweiler

[3] For information about GPT3/text-davinci-003 costs refer to: https://openai.com/api/pricing/

3 Modelli di diete sane e sostenibili a partire dalle diete tradizionali. Accordo di collaborazione tra Regione Umbria e ASL Città di Torino. [Dr. Andrea Pezzana, S.C. Nutrizione Clinica – ASL Città di Torino].
3.1 (paragrafi 1-2).
3.2 (paragrafi 2-3).
3.3 (paragrafo 3.1).
3.4 (paragrafo 4).
3.5 (paragrafo 4.1).
4 "Theoretical exploration for the combination of the ecological, energy, carbon, and water footprints: Overview of a footprint family" by Kai Fang, Reinout Heijungs, and Geert R. de Snoo.
5 "Sustainability of plant-based diets: back to the future" by Joan Sabate and Sam Soret.
6 "A multilevel carbon and water footprint dataset of food commodities" by Tashina Petersson, Luca Secondi, Andrea Magnani, Marta Antonelli, Katarzyna Dembska, Riccardo Valentini, Alessandra Varotto, and Simona Castaldi.
7 "Systematic review of greenhouse gas emissions for different fresh food categories" by Sarah Clune, Emma Crossin, and Karthik Natarajan Verghese.
8 "Working toward healthy and sustainable diets: the “Double Pyramid Model” developed by the Barilla Center for Food and Nutrition to raise awareness about the environmental and nutritional impact of foods" by L. F. Ruini et al.
9 https://huggingface.co/docs/transformers/training
10 "RoBERTa: A Robustly Optimized BERT Pretraining Approach" by Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov.
11 https://huggingface.co/sentence-transformers/all-MiniLM-L6-v2
12 https://colab.research.google.com/github/huggingface/blog/blob/main/notebooks/02_how_to_generate.ipynb
13 https://huggingface.co/EleutherAI/gpt-neo-2.7B
14 “Energy and Policy Considerations for Deep Learning in NLP” by Emma Strubell Ananya Ganesh Andrew McCallum.