Imperial College London

Machine Learning for Building a Food Recommendation System

Luís Artur Domingues Rita

MRes Biomedical Research (Data Science)

Supervisors: Prof. Kirill Veselkov

Prof. Michael Bronstein

09 March 2020

I certify that this thesis, and the research to which it refers, are the product of my own work, conducted during the current year of the MRes in Biomedical Research at Imperial College London. Any ideas or quotations from the work of other people, published or otherwise, or from my own previous work are fully acknowledged in accordance with the standard referencing practices of the discipline.

Luís Rita

Preface

The work presented in this thesis was performed at the Imperial College London between October 2019 and March 2020, under the supervision of Prof. Kirill Veselkov and Prof. Michael Bronstein.

Abstract

Many factors influence individual's health, such as physical exercise, sleep, nutrition, heredity and pollution. Being nutrition one of the biggest modifiable factors in our lives, small changes can have a big impact. With the exponential increase in the number of available food options, it is not possible to take them all into account anymore. The only way to consider user taste preferences, maximize the number of healthy compounds and minimize the unhealthy ones in food, is using (3D) recommendation systems.

The goal of this project was to use the largest publicly available collection of recipe data (Recipe1M+) to build a recommendation system for ingredients and recipes. Train, evaluate and test a model able to predict cuisines from sets of ingredients. Estimate the probability of negative recipe-drug interactions based on the predicted cuisine. Finally, to build a web application as a step forward in building a 3D recommendation system.

A vectorial representation for every ingredient and recipe was generated using Word2Vec. An SVC model was trained to return recipes' cuisines from their set of ingredients. South Asian, East Asian and North American cuisines were predicted with more than 73% accuracy. African, Southern European and Middle East cuisines contain the highest number of cancer-beating molecules. Finally, it was developed a web application able to predict the ingredients from an image, suggest new combinations and retrieve the cuisine the recipe belongs, along with a score for the expected number of negative interactions with antineoplastic drugs (github.com/warcraft12321/HyperFoods).

Machine Learning | Food Recommendation | Web Application

Acknowledgments

Truly grateful to my supervisor Prof. Kirill Veselkov that decisively contributed to the development of this project. And to Guadalupe Gonzalez that was always available to help.

☆ Alice ☆ Inas

Table of Contents

List of Figures

List of Tables

List of Acronyms

API	Application Programming Interface
JSON	JavaScript Object Notation
K&N	Kaggle and Nature
PCA	Principal Component Analysis
SVM	Support Vector Machine
SVC	Support Vector Classifier
T-SNE	T-Distributed Stochastic Neighbouring Entities
URL	Uniform Resource Locator

1. Introduction

There are many factors known that influences individual's health. Physical exercise, sleeping, nutrition, heredity, pollution, among other external factors [1]. Being nutrition one of the biggest modifiable factors in our lives, it is not surprising that small changes can induce significant outcomes [2].

Having our diets strong cultural ties, it is possible to identify around the world a big number of cuisines. The most common ingredients in each one is closely related to characteristics of the region, such as the climate. This plays a big influence in the availability of each of the components present in the local recipes [3].

Some molecules are known to have a positive effect in health, namely, in fighting cancer. Being able to identify which ingredients contain the higher concentrations, may help us treating and preventing the disease [4]. Moreover, by including these ingredients in tasty and affordable meals, it can promote a shift on the nutritional habits of the population. In a world where fast food consumption is rising, it is clear that additionally to the two previous points, speed of preparation is also an important factor [5].

With increasingly more data being made available online, whether from research studies or web applications, it is an opportunity to analyse it and create new food recommendation systems that not only take into account factors like anticancer properties, but also flavour, nutritional content and negative interactions with drugs. This would empower the user to take better decisions when buying or preparing his next meal [5].

1.1. Cancer

The XXI century disease (Figure 1). Belonging to a broader spectrum called tumours, cancers are a subtype where uncontrolled cell division occurs and has potential to spread to different tissues. In opposition, benign tumours are confined to a certain organ. Existing a close correlation between ageing and the loss of function of some of the regulatory pathways, as lifespan increases, the incidence of the disease is following the same trend [6].

It is known nutrition can play an important role in preventing and treating this disease [4]. This way, it would beneficial to maximize the number of cancer-beating compounds in food and minimize the ones known to interact negatively with anticancer drugs.

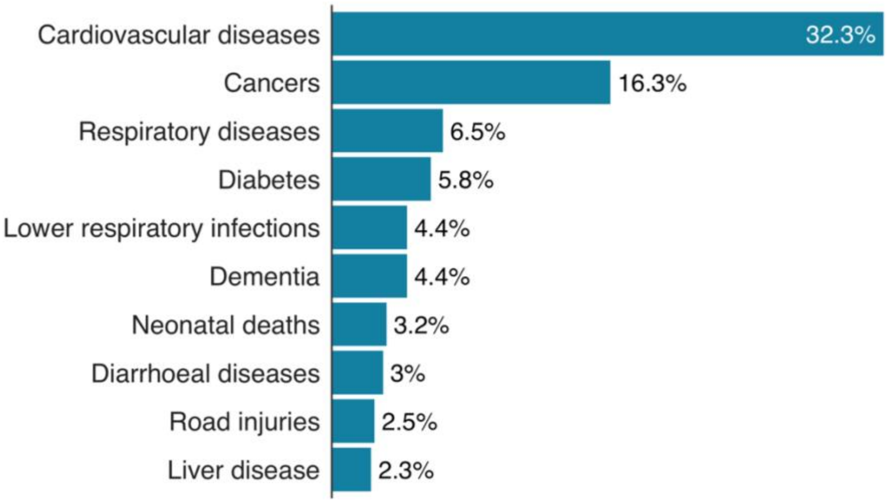


Figure 1 Leading causes of death worldwide, 2016. Cancer is the second in the list [7].

1.2. Natural Language Processing

Although online datasets and APIs (Application Programming Interface) contain structured information that can be easily retrieved, most online sources do not have such an organized structure. Consequently, algorithms that are able not only to extract data, but also to get its context are needed [8]. This section focuses in word embeddings and topic modelling.

There are several ways for achieving a vectorial representation for words. One possibility is to align them all and represent each one as a vector of 0s and the number 1 at the corresponding position of the alignment. Then, the dimensions of the vectorial space would be equal to the size of the vocabulary. Although this approach is feasible for small vocabularies, it is not computationally efficient. In alternative, there is a different word embedding mechanism that allows the representation of large vocabularies using low-dimensional vectors by accounting for the word’s context in the sentence.

Developed by a team of researchers led by Tomas Mikolov at Google, Word2Vec is a shallow, 2-layer neural network specific category of models that produces word embeddings [9]. It takes as input a corpus of text and spans a vectorial space, where each word is mapped into a vector. Words that more often appear in similar contexts are mapped into vectors separated by shorter Euclidean distances.

Word2Vec is not the only tool for topic modelling. *Doc2Vec* and *FastText* are able to encode whole documents or to look specifically at the morphological structure of each word, respectively [10].

Word2Vec was chosen to encode the ingredients, from the datasets used along this project, as vectors. This made possible to capture their similarities from their context in the recipes.

1.3. Inverse Cooking Algorithm

This recipe retrieval algorithm was developed by the Facebook AI Research and it is able to predict ingredients, cooking instructions and a title for a recipe, directly from an image [11].

In the past, algorithms have been using simple systems of recipe retrieval based on image similarities in some embedding space. This approach is highly dependent on the quality of the learned embedding, dataset size and variability. Therefore, these approaches fail when there is no match between the input image and the static dataset [11].

Inverse cooking algorithm instead of retrieving a recipe directly from an image, proposes a pipeline with an intermediate step where the set of ingredients is first obtained. This allows the generation of the instructions not only taking into account the image, but also the ingredients (Figure 2) [11].

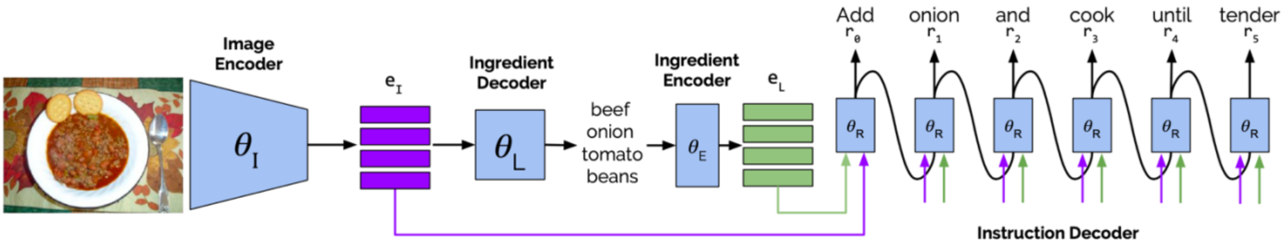


Figure 2 Inverse Cooking recipe generation model with the multiple encoders and decoders, generating the cooking instructions [11].

One of the major achievements of this method was to present higher accuracy than a baseline recipe retrieval system [12] and average human [11], while trying to predict the ingredients from an image.

Inverse Cooking algorithm was included in the food recommendation system developed in this project. Based on the predicted ingredients in the web application, several suggestions are provided to the user, such as: different ingredient combinations.

1.4. Dimensionality Reduction

Generally, dimensionality reduction aims to preserve as much information as possible from higher dimensional vectors. Principal Component Analysis (PCA) [13] and T-Distributed Stochastic Neighbouring Entities (T-SNE) [14] are two of the most commonly used approaches. The first is usually defined as having mathematical approach to the problem and the second a statistical one.

The main goal of PCA resides in preserving the vectorial components with higher variability across the data, while discarding the ones adding less information. This decomposition can be achieved in two different ways. One is by decomposing the data covariance matrix in its eigenvalues. The second is by performing Single Value Decomposition of the data matrix, after usually normalizing the initial data [13].

On the other hand, T-SNE converts the similarity between points into joint probabilities. And minimizes Kullback-Leibler divergence between these probabilities on the low-dimensional embedding and the high-dimensional data. This approach has a cost function that is not convex, consequently, different initializations may yield different dimensionality reduced vectors [14].

Being able to visualize high dimensional data is crucial particularly in the situation of being interested in performing clustering. Depending on the application, community finding algorithms can be input with different threshold parameters that influence the size and connectivity of the clusters. Being able to visualize how data is distributed allows for using human reasoning while choosing these values.

Its fast execution and reliable results, made PCA the first choice.

1.5. Community Finding

There are many clustering algorithms optimizing different cost functions.

Louvain algorithm iteratively partitions the network, optimizing modularity [15]. Its value is proportional to the number of connections among nodes inside the same cluster and decreases when the number of inter-cluster connections starts to rise. Mathematically, modularity is defined:

$$M = \frac{1}{2m}\sum_{i,j}(A_{ij} - p_{ij})\delta(c_i, c_j) = \frac{1}{2m}\sum_{i,j}(A_{ij} - \frac{k_i k_j}{2m})\delta(c_i, c_j) \qquad (1)$$

A_{ij} is the adjacency matrix entry representing the weight of the edge connecting nodes i and j, $k_i = \sum_j A_{ij}$ is the degree of node i, c_i is the community it belongs, δ-function $\delta(u, v)$ is 1 if $u = v$ and 0 otherwise. $m = \frac{1}{2}\sum_{ij} A_{ij}$ is the sum of the weights of all edges in the graph [15].

Between iteration steps, the value to be optimized is its variation which makes the calculations more efficient [15]:

$$\Delta M = \frac{k_{i,in}}{m} - \frac{2\sum_{tot} k_i}{(2m)^2} \Leftrightarrow \Delta Mm = k_{i,in} - \frac{\sum_{tot} k_i}{2m} \qquad (2)$$

While $k_{i,in}$ and Σ_{tot} need to be calculated for each trial community, $\frac{k_i}{2m}$ is specific of the node that is being analyzed. This way, the latter expression is only recalculated when a different node is considered while optimizing modularity [15].

Infomap algorithm attempts to reduce the description length of the network, by reducing an imaginary flow that propagates randomly inside the network between the identified clusters [16]. Mathematically it can be expressed:

$$L(M) = qH(Q) + \sum_{m=1}^{n_m} p_{\circledcirc}^{m} H(P_m) \qquad (3)$$

Being $q = \sum_{j=1}^{m} q_j$ the sum of the exit probability for each community, $H(Q)$ the average code length of the movement between communities, $p_{\circledcirc}^{i} = q_i + \sum_{\beta \in i}^{m} p_\beta$ the stay probability for a random walk in a community c and $H(P_m)$ the average code length of a module codebook m [16].

A third approach also widely used is called Spectral Clustering [17]. It has several advantages over the previous two. The most important is that it shows higher accuracy detecting clusters that are highly non-convex or when the measure of the centre and the spread of the cluster are not a suitable description of the predicted community. Moreover, contrarily to the other 2, its Python implementation allows the user to enter the number of desired clusters present in the data. This was the first choice to perform clustering along the project.

1.6. Data Visualization

There are several tools available for data visualization in Python. Namely, Matplotlib [18], Plotly [19], Seaborn [20] and Pandas [21]. Some are considered more efficient in representing larger quantities of data, others are known for being very versatile and allowing the user to easily visualize the data in multiple ways. Others are integrated with existent data containers and end up facilitating its visualization. Compatibility with the most recent platforms to analyse and visualize data, such as Jupyter Notebook or JupyterLab is also an important criterion.

Matplotlib allows the generation of a broad range of categories, such as: scatterplots, error charts, bar charts, power spectra, histograms, among others. Although, they are static and when the number of nodes in the case of a scatterplot goes beyond tens of thousands, it is not able to represent them anymore [18].

Plotly distinguishes from Matplotlib for allowing a dynamic representation of data points along with their labels. It also scales better when the magnitude of the number of points exceeds the tens of thousands [19].

Seaborn is a Python library built on the top of Matplotlib. It is powerful as the latter in representing a high number of datapoints and it allows the user to explore new visualization options in an easier way [20].

Due to the points highlighted before and the fact of being all compatible with JupyterLab (in the case of Plotly, after installing the respective extension) these 3 modules were used along the project.

1.7. Food Recommendation Systems

Although eating is a basic need, sometimes people do not know what to choose. In fact, when buying food or ordering online, the number of options is too big to be able to take them all into account.

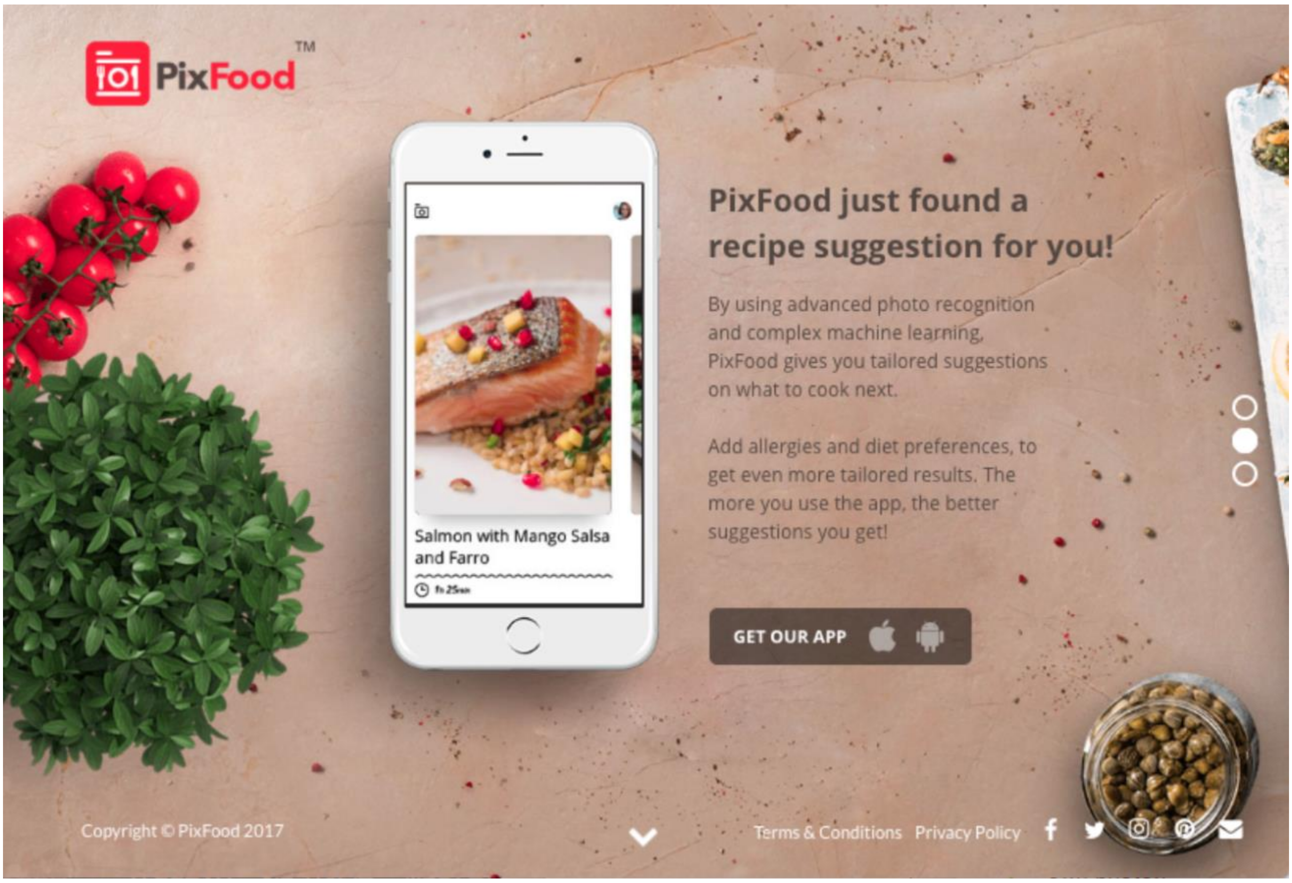
Humans have different nutritional needs and perceive flavour in different ways. For this reason, the only option to fulfil their needs is by knowing the person. Whether the recommendation is for a simple hungry user, cooking enthusiast, health concerned, dieter or someone ill looking for enhancing his medical status, this will influence the final choice [5].

Moreover, the product being recommended also has an important impact: a simple ingredient substitution, recipe, meal, restaurant or even a cuisine. The timing of the recommendation: real-time or as a newsletter. And it may take into consideration the location of the user and suggest the closest places. The platform where the suggestions are being made. Depending on how the recommendations are generated (collaborative filtering, content based, clustering of graphs or embeddings), they may require different features from the device. This way, they can be shared in a website, an application or as plain text (SMS). The presence of allergenic or intolerant compounds (such as nuts and milk), brands, required time for cooking, course, cuisines, presence of animal derivatives, type of dish, ingredients, cost, number of ingredients, preparation time, taste or the techniques required for cooking [5].

An important factor when building these systems is the source of the data. It can be from past orders, user's reaction to certain post (likes or dislikes), ratings attributed by the community, watched images or videos, or other social network related actions, including posts, shares, searches, comments or followers [5].

The success of a food recommendation system (Figure 3) is correlated with its ability to account for user preferences, maximize the number of healthy compounds and minimize the unhealthy ones in food.

Along this project, it was explored different approaches to generate food recommendations based on publicly available data. Large recipe datasets (Recipe1M+ and Kaggle and Nature (K&N)) containing information on the ingredients present, titles, source Uniform Resource Locators (URLs) and cuisines were parsed to provide the most precise recommendations.



PixFood just found a recipe suggestion for you!

By using advanced photo recognition and complex machine learning, PixFood gives you tailored suggestions on what to cook next.

Add allergies and diet preferences, to get even more tailored results. The more you use the app, the better suggestions you get!

Figure 3 An example of a food recommendation system [5].

1.8. Objectives

The aim of this project was to build a food recommendation system for ingredients and recipes. This included to pre-process Recipe1M+ dataset for ingredient retrieval. To optimize the vocabulary of ingredients to match them in the recipe text. To train a Word2Vec model able to convert ingredients and recipes into numerical vectors. To visualize ingredients in a 2D space and use it as an ingredient recommendation system. To train, evaluate and test a Support Vector Classifier (SVC) model able to predict the cuisine a recipe belongs, by considering its set of ingredients. Predict the probability of negative recipe-drug interactions based on the predicted cuisine. To identify the cuisines the higher average number of cancer-beating molecules per recipe. Finally, to create a web application able to predict the ingredients from an image, suggest new combinations and retrieve the cuisine the recipe belongs, along with a score for the expected number of negative interactions with antineoplastic drugs.

1.9. Thesis Outline

In Introduction, it was discussed the fundamental concepts behind Cancer, Natural Language Processing, Inverse Cooking Algorithm, Dimensionality Reduction, Community Finding, Data Visualization and Food Recommendation Systems available online. Methodology details the goals of the project and introduce the tools used to solve them. In Results, it is presented and discussed the outcomes of the project. Finally, in the Conclusion, the aim of the thesis is recalled, it is discussed whether the goals were achieved, main difficulties and suggestions for future work are provided.

2. Methodology

Introduction covered a broad range of theoretical concepts, particularly, the tools used along the project. In Methodology, it will be described how these were tuned and applied in accordance to the Objectives to be accomplished. The most important Python and JavaScript packages used in every step of the project were also included.

First, Recipe1M+ and K&N datasets, along with the ingredients' vocabulary, are described in terms of their structure and value for the project. It was also included the optimization processes they were submitted.

The process of embedding ingredients into a vectorial space, reduce their dimensionality and clustering are detailed in terms of the tools that were employed and the parameter tuning that was performed taking into account the characteristics of the previous datasets.

Next, the classifier that predicts the cuisines from each recipe set of ingredients was explained taking into consideration the choice of parameters and functions, accounting to the dimensions of the training set.

After, the approach to classify recipes and cuisines accordingly to their number of cancer-beating molecules and predicted number of negative recipe-drug interactions was described.

Data visualization tools used for each plot were specified.

Finally, a web application that retrieves the ingredients from a recipe's image and uses many of the food recommendation systems developed along this project is introduced and detailed in terms of its implementation.

2.1. Recipe1M+ Dataset

Recipe1M+ dataset is the biggest publicly available recipe dataset [22]. The information each recipe contains is separated in two JavaScript Object Notation (JSON) files.

The first identifies each recipe with an ID and defines the ingredients, instructions, title, URL, and the set it belongs: train, validate or test set. These three groups were used for training, validating and testing the Inverse Cooking algorithm. The second file includes the recipe IDs from the first and a set of URLs pointing towards images from the websites the recipes were scrapped. Some URLs are not active anymore. Although, using Wayback Machine (archive.org/web), it is possible to gain access and visualize the recipes.

Along with this dataset, two pickle files containing a vocabulary for the ingredients and instructions of the recipes were available.

One necessary step towards building a food recommendation system was to extract the ingredients from the recipes text in the Recipe1M+ dataset. To achieve this, it was optimized the existent ingredients' vocabulary where all the stop words and punctuation marks were removed, and the remaining words lemmatized.

Once Recipe1M+ dataset was scrapped from publicly available websites, containing recipes from its visitors, it is expected the presence of misinformation, typos, non-Latin characters, among others. It was verified, for example, that some recipes contain ingredients or instructions sets that were empty or exclusively formed by sequences of numbers or punctuation marks. It was corrected for all these cases. Before start searching the ingredients from the vocabulary in the dataset, all stop words were removed and the remaining lemmatized.

2.2. Kaggle and Nature Dataset

This dataset contains several recipes labelled with the cuisines they belong (github.com/altosaar/food2vec/blob/master/dat/kaggle_and_nature.csv). It was used to train the supervised learning model able to predict the cuisines from sets of ingredients. The same ingredients that were identified using the vocabulary described in the last section.

It is structured as a comma-separated file where each line contains a different recipe. The first value is the recipe's cuisine. The remaining are all the ingredients tokenized. For this reason, it was not needed the use of any vocabulary to retrieve them.

In order to coincide as much as possible the ingredients' designations in this dataset with the ones from Recipe1M+, all of them were lemmatized, and the stop words removed.

2.3. Food Embedding

In order to build an ingredient and recipe recommendation system, it was fundamental to represent them as vectors. This would allow to mathematically calculate their context similarities.

The Word2Vec model was trained using Recipe1M+ and K&N datasets. In the case of Recipe1M+ and K&N datasets, the ingredients present in each recipe were retrieved as explained in Recipe1M+ Dataset and Kaggle and Nature Dataset, respectively. Word2Vec tool is freely available in the python library *Gensim*.

One of the key points of training a Word2Vec model is capturing the surroundings of a word. Having this into consideration, it was fundamental to assure the order of the ingredients within each recipe followed a certain criterion. Otherwise, the model would interpret the same ingredients sorted differently as having a different context. For this reason, the ingredients were sorted alphabetically. The corpus introduced as input to the model was a set of sets of ingredients present in the recipes from the 2 datasets.

The hyperparameters that required appropriate tuning were the *size*, *workers*, *window*, *sg* and *min_count*. The *size* refers to the number of dimensions (100) considered for each vector representing the ingredients. The choice of this value was based on the quality of the obtained ingredient embeddings (Ingredient Recommendation) and accuracy of the model retrieving cuisines from sets of ingredients presented in the section Recipe to Cuisine. The number of *workers* was set equal to the number of cores (8) where the model was trained – MacBook Pro 15' Late 2016. This significantly speeded up the training process. The *window* refers to the maximum distance from the encoded word one wants to consider as still part of the surroundings. To calculate this value, the recipes with the highest number of ingredients in Recipe1M+ and K&N were identified. *window* was set to 65 after finding a recipe with 66 elements in K&N dataset. Finally, since the goal of the model was to predict a target vector from the neighbouring words, Continuous Bag of Words was chosen. This means, *sg* was introduced as 1. A final parameter that was considered was *min_count*. In order to obtain a vectorial representation for every ingredient, even if less represented, *min_count* was set to 1.

After obtaining ingredients' vectors, they were visualized in 2D plots. This was possible after reducing their dimensionality using the tool presented in the next section.

2.4. Dimensionality Reduction

To reduce the dimensionality of the ingredient embeddings created with Word2Vec from 100 to 2, it was used PCA. In *scikit-learn*, in the *decomposition* package, there are available several modules to reduce vectorial dimensions. One of them is *PCA*, which was used to allow the visualization of all the ingredients in the Recipe1M+ dataset.

2.5. Clustering Ingredients

In order to cluster the ingredients from Recipe1M+ dataset accordingly to their similarity, Spectral Clustering was applied to the ingredients' embeddings created with Word2Vec and dimensionality reduced with PCA. Again, it was used *scikit-learn*, but a different package. *cluster* includes many functions able to find communities in data, but the one that was used was *SpectralClustering*.

This function gives the user the possibility of inputting the desired number of clusters. It was chosen the same number as the number of ingredient categories that were identified in [3] – 9. They were main course, snack, beverage, soup/stew, bread, salad, appetizer, side dish, and dessert.

2.6. Recipe to Cuisine

An SVC model was trained to predict recipes' cuisines based on their sets of ingredients.

It was chosen K&N dataset to train the model because of its size, containing the list of ingredients in each recipe and the respective cuisine. Nonetheless, it was not possible to provide to the SVC a set of strings to perform the training. Each recipe was converted into a vector after averaging all the components of the ingredients' vectorial representations. The Word2Vec model presented before was trained with data from Recipe1M+ and K&N datasets, consequently, vectorial representations for every ingredient in the second dataset were already available.

In order to train the Support Vector Machine (SVM) model, it was used the SVM package from *scikit-learn*. It required the choice of a function, but also to tune several parameters accordingly to the training dataset.

Due to the high number of features (100), recipes and the computing limitations where the model was trained (MacBook Pro 15' Late 2016), it was used a linear kernel to decrease the training time. The function used was *LinearSVC*.

Several parameters of the function needed to be adjusted. K&N was found to be an unbalanced dataset. The size of some of the classes (cuisines) present in the dataset sometimes differ in two orders of magnitude. For this reason, *class_weight* was set to balanced. This means, it was attributed weights to each class inversely proportional to their number of elements. Next, to guarantee the training converges, it was increased the maximum number of iterations (*max_iter*) from the standard value of 1000 to 5000. Moreover, whenever datasets contain a higher number of elements than each recipe of features, it is advisable to set the algorithm to solve the dual optimization problem. This done by setting the *dual* parameter to False. There are some parameters (e.g.: regularization) that cannot be directly inferred from the characteristics of the dataset. For this reason, it was important to use *GridSearchCV* function from the *model_selection* package (also belonging to *scikit-learn*) so that an exhaustive search over a range of them could be made and the model's accuracy optimized. Values ranging from 0.00001 until 10000, in multiples of 10, were tested for the regularization parameter. 0.0001 was found to be optimal, then it was used to train the model. This parameter accounts for the importance that is given to the misclassification of data points.

2.7. Cuisine Classification

One of the goals of this project was to rank the cuisines accordingly to their number of cancer-beating molecules. First, it was determined the number of cancer-beating molecules in each recipe from the Recipe1M+ dataset, using the full set of ingredients in Table 1 (including the ones not represented) [4]. To match their names with the ingredients in the dataset, they were simplified. For example, *Common grape* was converted to *grape*. After retrieving the cuisines for each recipe in the dataset using the SVC introduced in

Recipe to Cuisine, it was calculated the average presence of cancer-beating molecules per recipe for each cuisine.

Table 1 Each ingredient is represented by its (modified) common and scientific names on the first and second columns, respectively. The number of cancer-beating molecules and its names are present in the last two. Only the five ingredients with the highest number of cancer-beating molecules were represented. Adapted from [4].

Common Name	Scientific Name	Nr. CBM	CBM Names
tea	*Camellia sinensis*	17	1,2,4-Trihydroxybenzene; 6-Keto-28-homobrassinolide; Apigenin; Brassinolide; Epigallocatechin 3-gallate; Gallic acid; Gallocatechin 3-gallate; Lupeol; Phloroglucinol; Procyanidin B2; Procyanidin B3; Prodelphinidin B4; Quercetin; Theaflavin; Tricetin; ?-Terpineol; ent-Epigallocatechin 3-gallate; ent-Gallocatechin 3-gallate
carrot	*Daucus carota*	12	Aesculetin; Apigenin; Carvone; Diosgenin; Ferulic acid 4-glucoside; Lupeol; Myristicin; Psoralen; Quercetin; Xanthotoxin; ?-Terpineol; ?-Elemene
grape	*Vitis vinifera*	12	Anthocyanidins; Betulinic acid; Epigallocatechin 3-gallate; Gallic acid; Gallocatechin 3-gallate; Lupeol; Procyanidin B1; Procyanidin B2; Procyanidin B3; Quercetin; ?-Terpineol; gamma-Tocotrieno
dill	*Anethum graveolens*	12	(R)-Carvone; (S)-Carvone; Aesculetin; Apigenin; Apiole; Carvone; Myristicin; Quercetin; Umbelliprenin; Xanthotoxin; ?-Terpineol; ?-Elemene
celery	*Apium graveolens*	12	8-p-Menthene-1,2-diol; Angelicin; Apigenin; Apiole; Carvone; Myristicin; Psoralen; Quercetin; Verbenol; Xanthotoxin; ?-Terpineol; ?-Elemene

Cuisines can also be classified accordingly to the expected number of negative interactions their recipes have with drugs. For the case of Antineoplastic and Immunomodulating Agents, the permillage of expected harmful interactions was imported (Table 2) [3]. This information is going to be used when predicting the probability for negative recipe – drug interactions after retrieving the cuisine from a recipe's image using the HyperFoods App.

Table 2 Permillage of negative interactions expected between anticancer drugs and the cuisines around the globe [3]. Abbreviations: NA (North American), WE (Western European), NE (Northern European), EE (Eastern European), SE

(Southern European), ME (Middle Eastern), SA (South Asian), SEA (Southeast Asian), EA (East Asian), LA (Latin American) and A (African).

Drugs	NA	WE	NE	EE	SE	ME	SA	SEA	EA	LA	A
Antineoplastic and Immunomodulating Agents	4.430	4.673	3.818	4.156	2.744	1.644	1.720	1.276	0.844	2.137	1.442

2.8. Visualization Tools

Matplotlib, Plotly and Seaborn were the three python visualization frameworks chosen to visualize some of the project's data.

Matplotlib was used to plot all the 2D ingredient vectorial embeddings in the report. In the Jupyter Notebook available in the project's repository, it is possible to use Plotly to visualize the same data dynamically. Making it easier to inspect ingredient by ingredient, once their labels were initially hidden, only appearing when hovering the respective nodes with the mouse. It is also possible to zoom in and out to get more or less detail, respectively.

Seaborn plotted the confusion matrix that resulted from evaluating the cuisine retrieval algorithm detailed in Recipe to Cuisine.

2.9. HyperFoods App

As a step forward in building a new recommendation platform for ingredients and recipes, it was developed a web application. This is able to predict the ingredients from an image (provided by an URL). To suggest alternative ingredients based on their proximity in the embedding vectorial space to the ones identified by the Inverse Cooking algorithm. To predict a cuisine from the retrieved set of ingredients. And estimate the probability of negative recipe- antineoplastic drug interactions based on the predicted cuisine.

The implementation of the backend and frontend of the web application are described below.

The backend was developed using Node.js. In the server-side, it was imported the file that contains the top 3 most similar ingredients from all in the vocabulary. This file was obtained after calculating the Euclidean distance among all vectors in the embedding space and exporting the resulting dictionary as a JSON file. It was established a connection between the frontend HTML page and the server, by setting a listening port. A python shell was used to: execute the Inverse Cooking algorithm which is implemented in Python; load the Word2Vec model; convert images' ingredients to vectors; calculate a vectorial representation for each recipe and load the SVC model to be able to predict the cuisine from the recipe vector. A function to convert binary code back into a string was implemented and executed in the server-side. Due to the format of a general URL, it was not possible to fetch the link from the frontend to the server. So, this was converted into binary in the frontend and back to a string in the backend.

In the frontend, it was used HTML to build the main structure of the interface. CSS was employed to make the interface simple and intuitive to use. And, JavaScript, enabled a responsive webpage and the communication with the server. One fundamental JavaScript library that was used to make the website user-friendly was the minified version of D3.js, version 4. It handled all the mouse events.

After implementing the backend and frontend, the goal was to make the web application available online. Due to the size limits of Heroku App (hosting platform) and the size of some PyTorch files this was not achieved. While executing the Inverse Cooking algorithm, it was used a version of the PyTorch module that included GPU and CPU support. The module occupied in memory more than 1GB. Due to storage limitations (500 MB) of the hosting platform, it was used a lighter version of PyTorch executing only on the CPU. This would not impact the execution of the application, once Heroku does not provide graphical processing power. This modification reduced the size of the application in half. But there was still the trained PyTorch model from the Inverse Cooking algorithm that was causing the app to exceed server's memory limits. One way to overcome this would be to retrain the model using a smaller training dataset, but this would significantly decrease the accuracy of detection. Due to time constraints, this path was not followed. The size of the model (436 MB) was not enough to exceed the threshold value but, in parallel with the Node.js and Python packages that were required for the execution of the app, it was not possible to make it available online.

3. Results

In Methodology, the goals of the project were detailed along with the tools employed to achieve them. In Results, the outcomes from data analysis and visualization are presented. Along with a discussion about the predictability of the results and their impact.

First, it is provided a detailed overview on the datasets that were used for data analysis (Recipe1M+ Dataset) and model training (Kaggle and Nature Dataset). Then, the ingredient embeddings and criteria to visualize them in a 2-dimensional space are specified (Ingredient Recommendation). Next, the performance of the SVC is evaluated using a confusion matrix and tested afterwards (Recipe to Cuisine). Cuisines with the higher number of cancer-beating molecules per recipe are ranked (Cuisine Classification). Finally, it is presented the food recommendation web application that was developed (HyperFoods App).

3.1. Recipe1M+ Dataset

This dataset contains 1029715 recipes which are compound by 1480 different ingredients.

In order to get a better overview on the source of the recipes, the dataset was parsed, and a list of the crawled websites returned (Table 3). Most of the databases are European or American. This way, it was expected the cuisine retrieval algorithm to classify a big number of recipes into these categories.

Table 3 Scrapped websites used to create Recipe1M+ dataset on the left. In the middle, the respective URLs. On the right, the number of recipes from each source.

Dataset Name	URL	N. Recipes
Food.com	food.com	509998
Tasty Kitchen	tastykitchen.com	75548
Cookpad	cookpad.com	61636
CookEatShare	cookeatshare.com	60628
Food Network	foodnetwork.com	58156
Kraft Recipes	kraftrecipes.com	50850
Allrecipes	allrecipes.com	49101
Epicurious	epicurious.com	48723
RecipeLand.com	recipeland.com	27332

Food & Wine Magazine	foodandwine.com	18273
NYT Cooking	cooking.nytimes.com	17453
Foodgeeks	foodgeeks.com	10317
Cookstr	cookstr.com	9240
MyRecipes	myrecipes.com	7153
Chowhound	chowhound.com	6361
The On-Line Cookbook	online-cookbook.com	5764
Vegetarian Times	vegetariantimes.com	4792
Delish.com	delish.com	4170
Land O'Lakes	landolakes.com	2562
Food Republic	foodrepublic.com	2341
Lovefood.com	lovefood.com	1940

In order to retrieve the ingredients from Recipe1M+, it was developed an optimized version of the vocabulary that was created by the same team that released the dataset.

Table 4 shows the top 5 ingredients most often retrieved from the recipes in the Recipe1M+ dataset and the respective number of occurrences. This data was used while plotting the ingredients in the 2D embedding vectorial space and it was discussed its relevance in terms of ingredient recommendations (Ingredient Recommendation).

Table 4 Top 5 ingredients present most often in the recipes from Recipe1M+ dataset on the left. On the right, the respective number of occurrences.

Ingredients	Occurrences
salt	593043
pepper	561840
sugar	464057
oil	402961
onion	362011

3.2. Kaggle and Nature Dataset

K&N dataset contains 96250 recipes and 3904 different ingredients across 11 cuisines: North American, Western European, Northern European, Eastern European, Southern European, Middle Eastern, South Asian, Southeast Asian, East Asian, Latin American and African. Although this dataset contains approximately 10% of the number of recipes in Recipe1M+, it contains more than the double on the number of different ingredients. This happens because simple ingredients such as sugar, in this dataset, are separated in different ones depending on their colour or origin (e.g.: organic granulated sugar, superfine white sugar, baking sugar…).

In Table 5 were represented the top 5 most common ingredients. The most common ingredients in Recipe1M+ and K&N datasets significantly overlap. In the top 5, onion and pepper are present in both datasets.

Table 5 On the left, ingredients occurring more often in K&N dataset. On the right, the respective number of presences.

Ingredients	Occurrences
garlic	35804
pepper	32979
onion	32893
butter	29043
egg	26875

In Table 6, it is represented the distribution of cuisines in the dataset, in terms of the number of recipes. There is an unbalance on the number of recipes per cuisine detected in the K&N dataset. North American cuisine contains 2 orders of magnitude more recipes than Eastern European. This will influence SVC model accuracy (Confusion Matrix) when predicting cuisines from the list of ingredients.

Table 6 Number of recipes belonging to the different cuisines in the K&N dataset. Abbreviations: NA (North American), WE (Western European), NE (Northern European), EE (Eastern European), SE (Southern European), ME (Middle Eastern), SA (South Asian), SEA (Southeast Asian), EA (East Asian), LA (Latin American) and A (African).

	NA	WE	NE	EE	SE	ME	SA	SEA	EA	LA	A	Total
Nr. Recipes	45843	6774	739	381	14178	645	3618	3572	7435	11892	1173	96250

3.3. Ingredient Recommendation

After representing each ingredient present in Recipe1M+ dataset in 100-features embedding and reduce dimensionality to a bidimensional space, the plot in Figure 4 was obtained. For clarity, all ingredients occurring less than 800 times in the dataset were not represented. Different colours correspond to different clusters that were identified using spectral clustering.

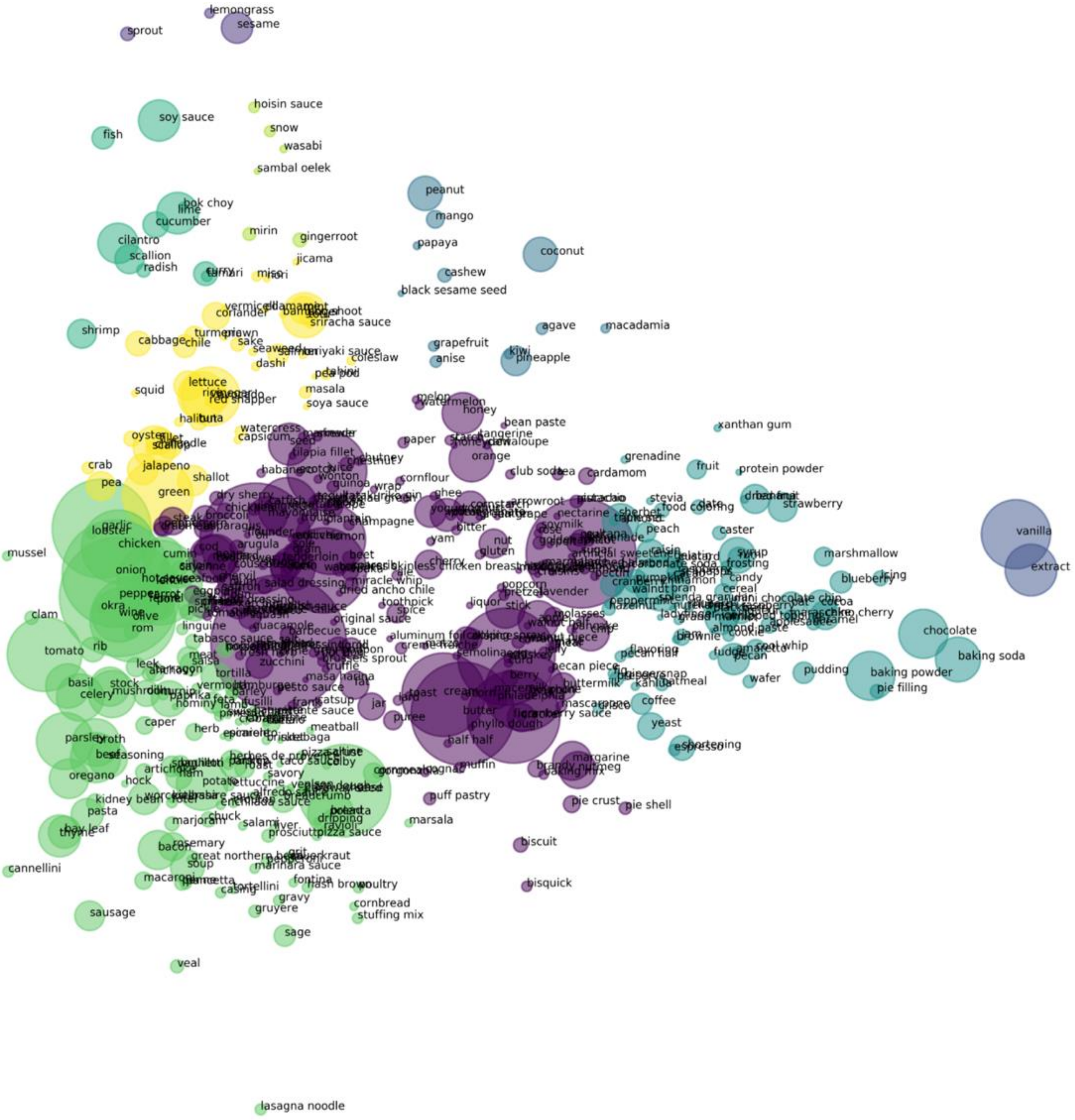


Figure 4 Context similarity for several ingredients in Recipe1M+. Only represented the ones with at least 800 occurrences in the dataset. Ingredients coloured accordingly to the cluster they belong.

This food representation allows us to extract information regarding what are the ingredients that most often co-occur. Consequently, it offers a baseline for trying new combinations based on their proximity in the plot. As bigger is the radius of the overlapping ingredients, as bigger is the confidence of the success of their combination. Some examples are garlic, chicken and onion; tomato, basil and celery or, the unexpected, honey and orange.

The same image was reprinted, but with a different colouring criterion. This time, ingredients containing at least one cancer-beating molecule were coloured green and the remainings black (Figure 5).

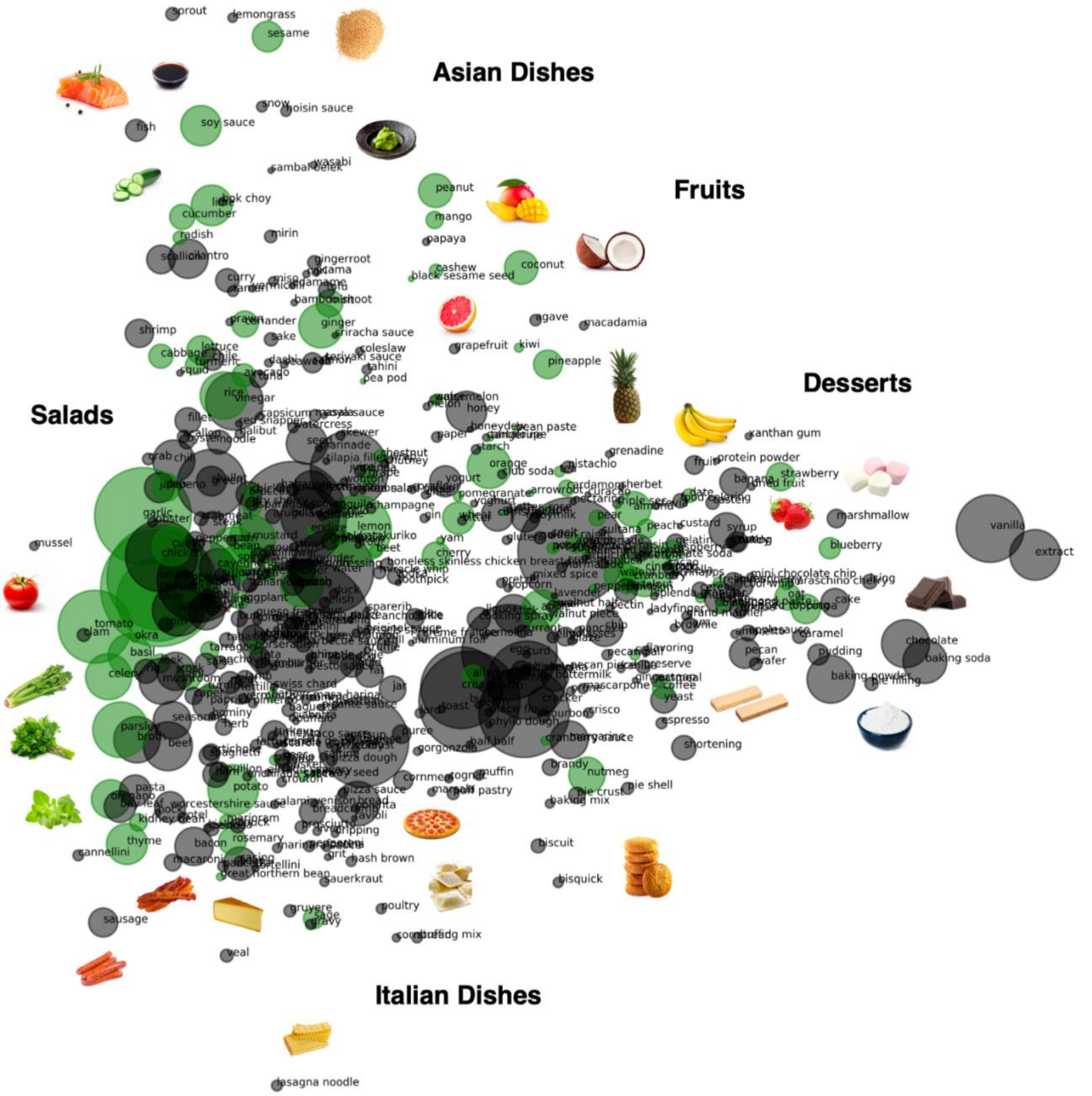


Figure 5 Context similarity of the ingredients in Recipe1M+. Only represented the ones with at least 800 occurrences in the dataset. Green nodes (ingredients) contain at least 1 cancer-beating molecule.

By identifying the ingredients that are closer to the ones containing cancer-beating molecules (green nodes), one may hypothesize that given the similar contexts they are present in the recipes, it might be more likely to find these same molecules on those than in any others.

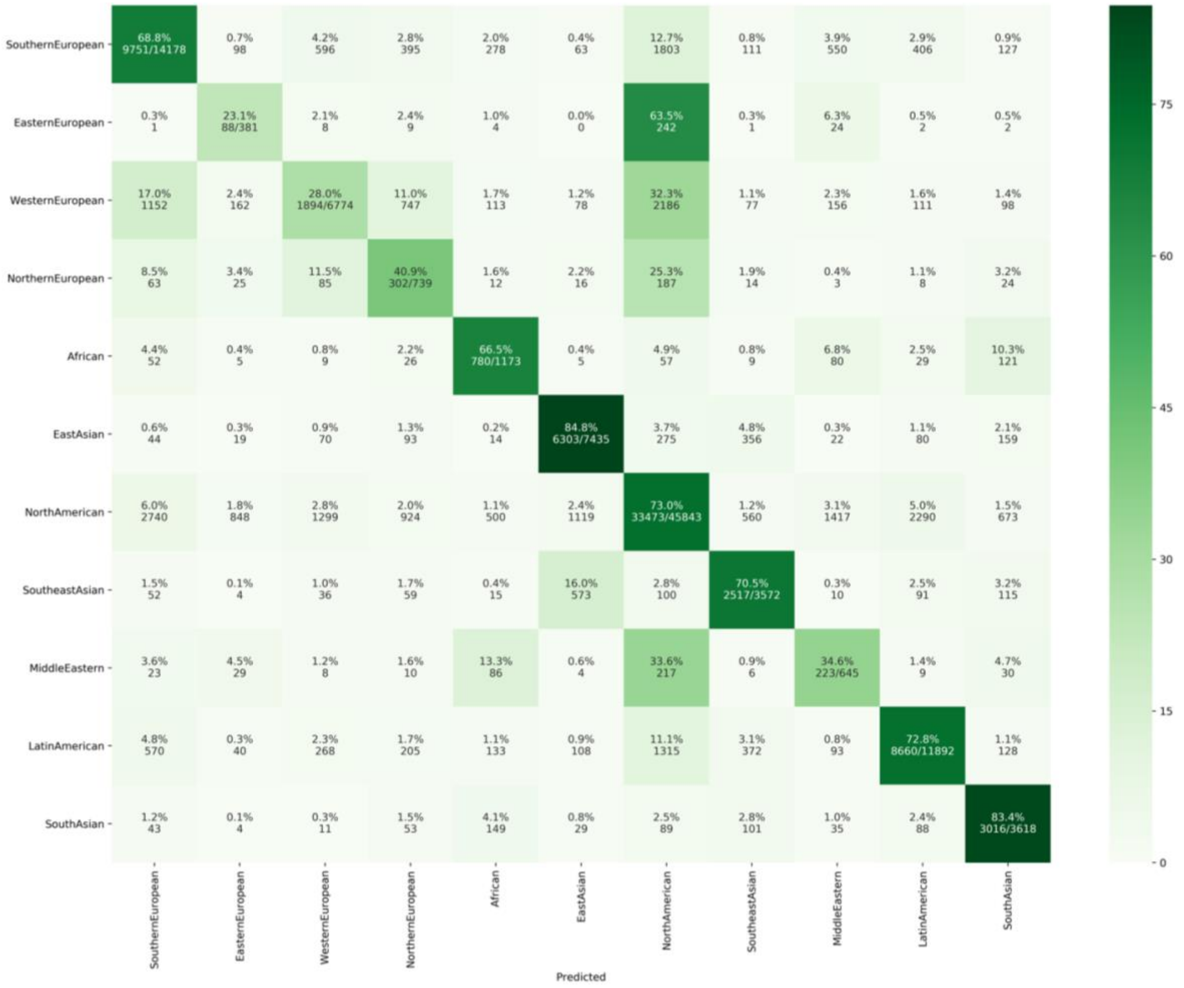
Moreover, Figure 5 gives us information not only on the ingredients that usually co-occur, but also provides the user information on their content in cancer-beating molecules. This way, the user can take into account their anticancer properties, while considering their context similarity.

3.4. Recipe to Cuisine

It was discussed in Recipe to Cuisine, the implementation of the SVC that is able to predict the cuisines, given a set of ingredients contained by the recipe datasets Recipe1M+ and K&N. In this section, its accuracy is assessed after building a confusion matrix, the model is tested in Recipe1M+ dataset and some of the ingredients that play an important role in cuisine classification are identified.

3.4.1. Confusion Matrix

In order to understand how different cuisines are detected by the classifier, a multidimensional confusion matrix was calculated from the training set (Figure 6). It includes the prediction for each one of the 11 cuisines in the K&N recipe dataset.



	SouthernEuropean	EasternEuropean	WesternEuropean	NorthernEuropean	African	EastAsian	NorthAmerican	SoutheastAsian	MiddleEastern	LatinAmerican	SouthAsian
SouthernEuropean	68.8% 9751/14178	0.7% 98	4.2% 596	2.8% 395	2.0% 278	0.4% 63	12.7% 1803	0.8% 111	3.9% 550	2.9% 406	0.9% 127
EasternEuropean	0.3% 1	23.1% 88/381	2.1% 8	2.4% 9	1.0% 4	0.0% 0	63.5% 242	0.3% 1	6.3% 24	0.5% 2	0.5% 2
WesternEuropean	17.0% 1152	2.4% 162	28.0% 1894/6774	11.0% 747	1.7% 113	1.2% 78	32.3% 2186	1.1% 77	2.3% 156	1.6% 111	1.4% 98
NorthernEuropean	8.5% 63	3.4% 25	11.5% 85	40.9% 302/739	1.6% 12	2.2% 16	25.3% 187	1.9% 14	0.4% 3	1.1% 8	3.2% 24
African	4.4% 52	0.4% 5	0.8% 9	2.2% 26	66.5% 780/1173	0.4% 5	4.9% 57	0.8% 9	6.8% 80	2.5% 29	10.3% 121
EastAsian	0.6% 44	0.3% 19	0.9% 70	1.3% 93	0.2% 14	84.8% 6303/7435	3.7% 275	4.8% 356	0.3% 22	1.1% 80	2.1% 159
NorthAmerican	6.0% 2740	1.8% 848	2.8% 1299	2.0% 924	1.1% 500	2.4% 1119	73.0% 33473/45843	1.2% 560	3.1% 1417	5.0% 2290	1.5% 673
SoutheastAsian	1.5% 52	0.1% 4	1.0% 36	1.7% 59	0.4% 15	16.0% 573	2.8% 100	70.5% 2517/3572	0.3% 10	2.5% 91	3.2% 115
MiddleEastern	3.6% 23	4.5% 29	1.2% 8	1.6% 10	13.3% 86	0.6% 4	33.6% 217	0.9% 6	34.6% 223/645	1.4% 9	4.7% 30
LatinAmerican	4.8% 570	0.3% 40	2.3% 268	1.7% 205	1.1% 133	0.9% 108	11.1% 1315	3.1% 372	0.8% 93	72.8% 8660/11892	1.1% 128
SouthAsian	1.2% 43	0.1% 4	0.3% 11	1.5% 53	4.1% 149	0.8% 29	2.5% 89	2.8% 101	1.0% 35	2.4% 88	83.4% 3016/3618

Figure 6 Confusion matrix for the SVM model trained using K&N dataset. Each row, along with the respective values, represent the proportion of ingredients correctly assigned to the real cuisine. Each column is a cuisine that was predicted by the algorithm.

Recipes belonging to the Northern European, Eastern European, Western European and Middle Eastern regions were often misclassified as North American. In the case of Northern European, Eastern European and Middle East, it may be due to their underrepresentation in the dataset: 739, 381 and 645 recipes, respectively. As well as, overrepresentation of North American cuisine.

Another important factor are the common ingredients. In fact, all European cuisines are often misclassified among them. As an example, 11.5% of the Northern European recipes were misclassified as Western European. Europe is the continent represented with the highest number of cuisines (4). This may be reducing specificity among them.

The model was more accurate in predicting East Asian, South Asian and North American recipes. In fact, it did it correctly for 84.8%, 83.4% and 73.0% of them, respectively.

3.4.2. Testing Classifier

After training the classifier, it was used to predict the cuisines of every recipe in the Recipe1M+ dataset. The respective proportions were depicted in Table 7.

In order to test the classifier, it was used the following approach: first, all recipes with the word *tea* present in the title were retrieved. Then, it was calculated what was the final cuisine distribution (Table 7).

Table 7 Distribution of cuisines in Recipe1M+ dataset and the analogous by considering only recipes which title contain the keyword *tea*.

	NA	WE	NE	EE	SE	ME	SA	SEA	EA	LA	A	Total
Tea Recipes	526 12%	248 6%	1073 25%	56 1%	228 5%	49 1%	209 5%	222 5%	1140 27%	123 3%	338 8%	4212 100%
Total Recipes	139731 14%	40303 4%	162329 16%	21818 2%	218260 21%	6383 1%	20979 2%	131668 13%	102875 10%	149331 15%	35679 3%	1029356 100%

As expected, the majority of recipes containing the word tea in the title were classified as belonging, first, to East Asian (27% recipes) and, second, to North European (25% recipes) cuisines (Figure 7). These two areas are much more represented in the tea selected recipes than in the overall Recipe1M+ dataset. In East Asia, China is the largest absolute consumer of tea in the world, at 1.6 billion pounds a year [23]. From Northern European countries, Ireland, United Kingdom and Russia are in the top 4 countries with higher consumption rates per capita [23]. On the other hand, the highest ranked South European country in the list (Spain) only appears in 40th place on tea consumption per capita [23]. And, in fact, the largest drop in the dataset (from 21% to 5%) was verified for this region.

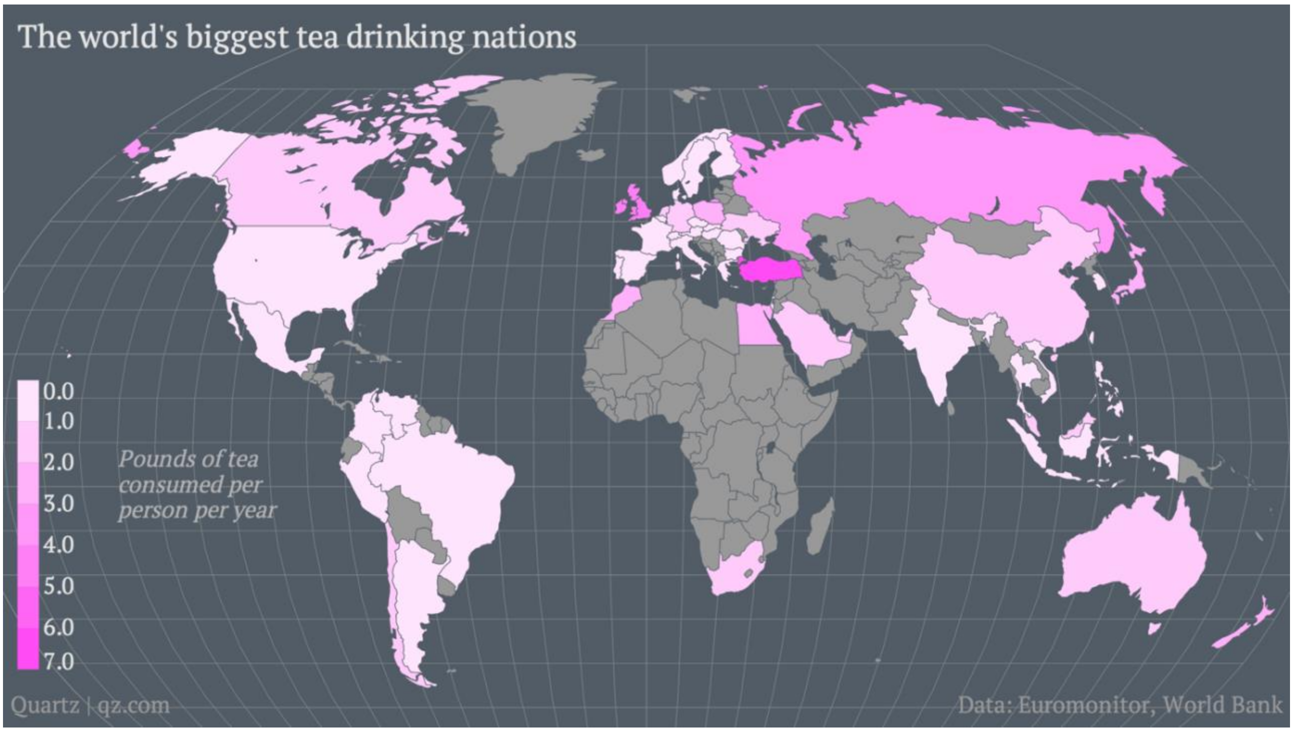

The world's biggest tea drinking nations
0.0
1.0
2.0
3.0
4.0
5.0
6.0
7.0
Pounds of tea consumed per person per year
Quartz | qz.com
Data: Euromonitor, World Bank

Figure 7 Consumption rates of tea around the world per person per year [23].

3.5. Cuisine Classification

In order to analyse the presence of cancer-beating molecules among the different cuisines, it was used the information on the number of cancer-beating molecules in each recipe. After averaging this number over all the recipes within a cuisine category, it was determined a score for each one (Table 8).

Table 8 Average number of cancer-beating molecules per recipe for each of the 11 represented cuisines. Abbreviations: NA (North American), WE (Western European), NE (Northern European), EE (Eastern European), SE (Southern European), ME (Middle Eastern), SA (South Asian), SEA (Southeast Asian), EA (East Asian), LA (Latin American) and A (African).

	NA	WE	NE	EE	SE	ME	SA	SEA	EA	LA	A
Anticancer Score	1.25	1.49	1.30	1.90	2.21	2.08	1.90	1.25	1.32	1.75	2.48

The highest scores were obtained for three regions in the Mediterranean area: South Europe, Middle East and Africa (Figure 8). On the opposite side, North American cuisine was identified with the lowest number of cancer-beating molecules. These results are in line with predictions to the North American cuisine which is often classified as unhealthy and positively correlated with the incidence of oncologic diseases [24]. And the Mediterranean diet which is rich in fruits and vegetables and often identified as cancer preventive [25].

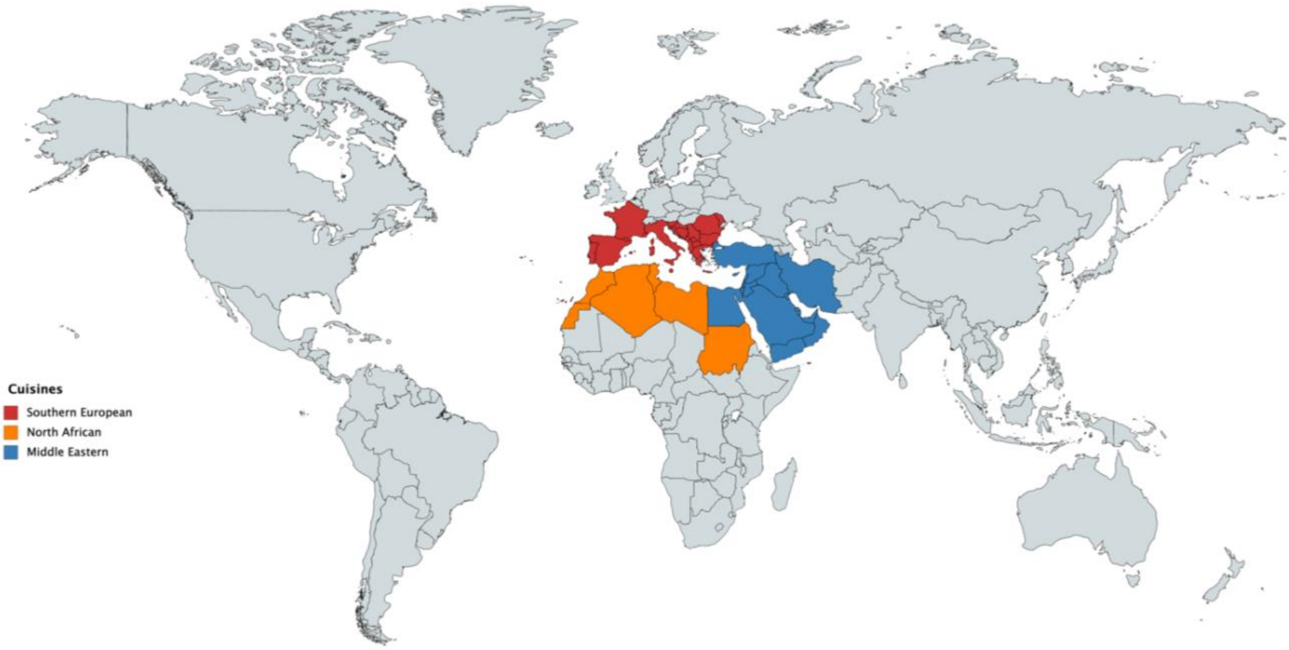


Figure 8 Cuisines containing recipes with the higher number of cancer-beating molecules. World map generated using [26].

In order to understand which ingredients could be responsible for making South European, Middle Eastern and African cuisines highly ranked, it was calculated their distribution by considering only the ones containing the keyword *salad* in their title (Table 9).

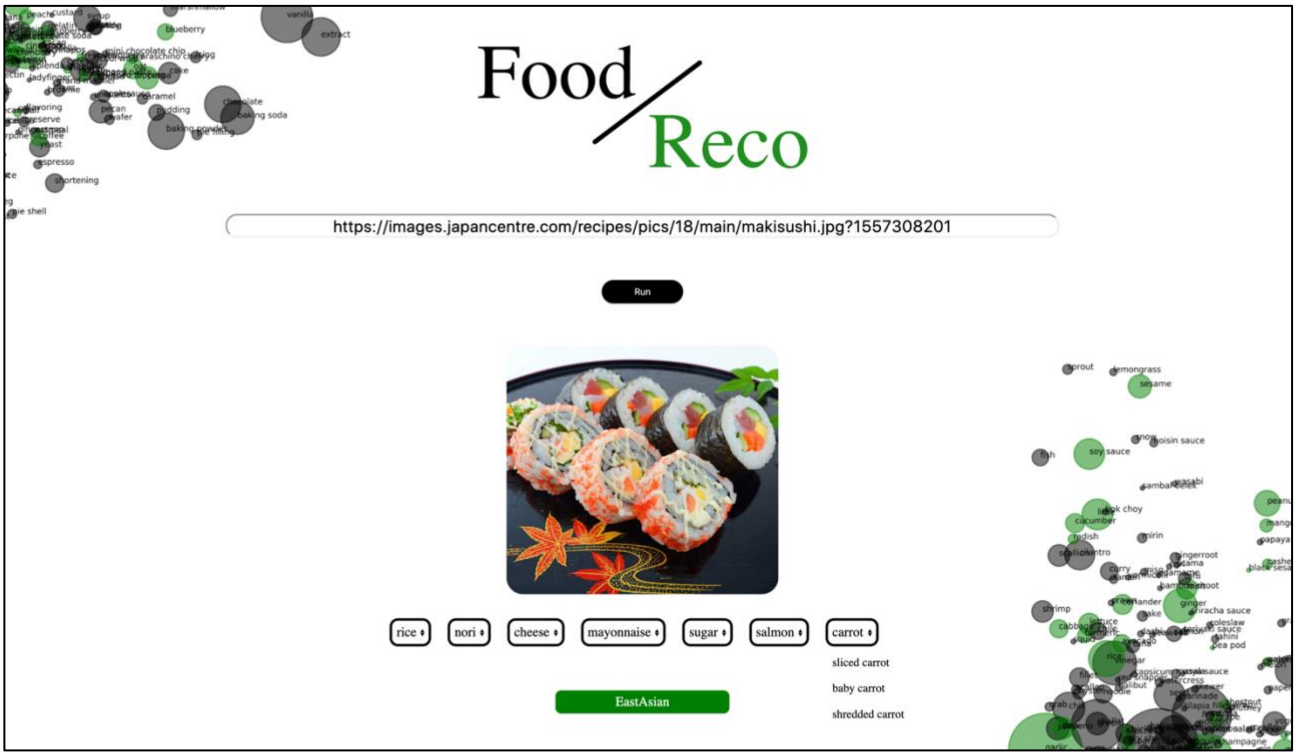
Table 9 Distribution of cuisines in Recipe1M+ dataset and the analogous by considering only recipes which title contain the keyword *salad*. Abbreviations: NA (North American), WE (Western European), NE (Northern European), EE (Eastern European), SE (Southern European), ME (Middle Eastern), SA (South Asian), SEA (Southeast Asian), EA (East Asian), LA (Latin American) and A (African).

	NA	WE	NE	EE	SE	ME	SA	SEA	EA	LA	A	Total
Salad Recipes	12739 20%	349 1%	3867 6%	1793 3%	22577 36%	764 1%	730 1%	3673 6%	5425 9%	8954 14%	2058 3%	62929 100%
Total Recipes	139731 14%	40303 4%	162329 16%	21818 2%	218260 21%	6383 1%	20979 2%	131668 13%	102875 10%	149331 15%	35679 3%	1029356 100%

Table 9 shows that salad recipes were mostly classified as part of the Southern European cuisine. They might be one of the main responsible for increasing the anticancer score of this cuisine, once the majority of the ingredients present in the complete version of Table 1 are commonly found in salads. Additionally, it is noteworthy to highlight that one of the main components of the South European diet is salad [27].

3.6. HyperFoods App

On the top of the webpage, FoodReco receives an URL of an image available online and returns a list of predicted ingredients after executing the Inverse Cooking algorithm. They will be displayed below the image that was processed. By hovering each ingredient with the mouse, an ordered list of the top 3 ingredients that most often co-occur in Recipe1M+ and K&N is presented. On the bottom is displayed the predicted cuisine, based on the set of ingredients. The background colour of this text field is green whenever the number of negative interactions with drugs is below average or red if above for antineoplastic drugs (Figure 9). This value is calculated after averaging all permillages from Table 2.



https://images.japancentre.com/recipes/pics/18/main/makisushi.jpg?1557308201

Figure 9 Interface of FoodReco web application. It is able to predict the ingredients from an image, to suggest 3 ingredient substitutions for each one and retrieve the most likely cuisine the dish belongs.

4. Conclusion

The goal of this project was to use the largest publicly available collection of recipe data (Recipe1M+) to build a recommendation system for ingredients and recipes. Train, evaluate and test a model able to predict cuisines from sets of ingredients. Predict the probability of negative recipe-drug interactions based on the predicted cuisine. Finally, to build a web application as a step forward in building a recommendation system which accounts for the user taste preferences, maximizes the number of healthy compounds and minimizes the unhealthy ones in the food.

A vectorial representation for every ingredient was generated using Word2Vec. By averaging all the components of the vectorized ingredients, it was obtained a similar representation for recipes. An SVC model was trained to return recipes' cuisines from their set of ingredients. South Asian, East Asian and North American cuisines were predicted with more than 73% accuracy. African, Southern European and Middle East cuisines contain the highest number of cancer-beating molecules. Finally, it was developed a web application able to predict the ingredients from an image, suggest new combinations and retrieve the cuisine the recipe belongs, along with a score for the expected number of negative interactions with antineoplastic drugs. It was not possible to make it available online, but it can be executed locally (github.com/warcraft12321/HyperFoods).

As future enhancements to the approach used in this project are the creation of a more exhaustive and accurate vocabulary for ingredients. Before embedding ingredients using Word2Vec, the uniformization between the ones present in the Recipe1M+ and K&N datasets could enhance the accuracy of the embeddings and of the SVC. Generating vectors for the recipes accounting for the proportion of the different ingredients. One step in this direction was taken in this project by developing a vocabulary including all the units (customary and from usual metric systems) present in the Recipe1M+ dataset. As well as, a conversion system between all the identified units to grams. Different kernels other than the linear could have been used to train the cuisine retrieval SVC model. Or, deep learning employed to optimize the same problem. Finally, although the web application runs smoothly in a local computer, next step would be to make it available online and with an enhanced number of features, such as: adding additional food recommendations for selecting ingredients rich in cancer-beating molecules (HyperFoods), while decreasing the number of negative drug-cuisine interactions for other types of drugs other than the antineoplastic.

References

[1] H. Arem and E. Loftfield, “Cancer Epidemiology: A Survey of Modifiable Risk Factors for Prevention and Survivorship,” *American Journal of Lifestyle Medicine,* vol. 12, no. 3, p. 200–210, 2018.

[2] M. S. Donaldson, “Nutrition and cancer,” *Nutrition Journal,* vol. 3, pp. 19-25, 2004.

[3] M. Jovanovik, A. Bogojeska and D. e. a. Trajanov, “Inferring Cuisine- Drug Interactions Using the Linked Data Approach,” *Scientific Reports,* vol. 5, no. 9346, 2015.

[4] K. Veselkov, G. Gonzalez, S. Aljifri, D. Galea, R. Mirnezami, J. Youssef, M. Bronstein and I. Laponogov, “HyperFoods: Machine intelligent mapping of cancer-beating molecules in foods,” *Scientific Reports,* vol. 3, no. 9237, 2019.

[5] C. Anderson, “A survey of food recommenders,” *ArXiv,* vol. abs/1809.02862, 2018.

[6] J. R. Aunan, W. C. Cho and K. Søreide, “The Biology of Aging and Cancer: A Brief Overview of Shared and Divergent Molecular Hallmarks,” *Aging and disease,* vol. 8, no. 5, p. 628–642, 2017.

[7] “IHME, Global Burden of Disease, Our World in Data,” 2016. [Online]. Available: http://www.healthdata.org/gbd. [Accessed 8 March 2020].

[8] M. Chary, S. Parikh, A. F. Manini, E. W. Boyer and M. Radeos, “A Review of Natural Language Processing in Medical Education,” *The Western Journal of Emergency Medicine,* vol. 20, no. 1, 2019.

[9] T. Mikolov, K. Chen, G. Corrado and J. Dean, “Efficient Estimation of Word Representations in Vector Space,” *Proceedings of the International Conference on Learning Representations,* 2013.

[10] R. Řehůřek, “gensim: Topic modelling for humans,” [Online]. Available: https://radimrehurek.com/gensim/. [Accessed 8 March 2020].

[11] A. Salvador, M. Drozdzal, X. Giro-i-Nieto and A. Romero, “Inverse Cooking: Recipe Generation from Food Images,” *Computer Vision and Pattern Recognition,* 2018.

[12] A. Salvador, N. Hynes, Y. Aytar, J. Marin, F. Ofli, I. Weber and A. Torralba, “Learning cross-modal embeddings for cooking recipes and food images,” *Computer Vision and Pattern Recognition,* 2017.

[13] I. T. Jolliffe and J. Cadima, “Principal component analysis: a review and recent developments,” *Philosophical Transactions of The Royal Society A Mathematical Physical and Engineering Sciences,* vol. 374, no. 2065, 2016.

[14] L. v. d. Maaten and G. Hinton, “Visualizing Data using t-SNE,” *Journal of Machine Learning Research,* vol. 9, pp. 2579-2605, 2008.

[15] V. D. Blondel, J.-L. Guillaume, R. Lambiotte and E. Lefebvre, “Fast unfolding of communities in large networks,” *J. Stat. Mech.,* 2008.

[16] M. Rosvall, D. Axelsson and C. T. Bergstrom, “The map equation,” *The European Physical Journal Special Topics,* vol. 178, no. 1, pp. 13-23, 2009.

[17] A. Y. Ng, M. I. Jordan and Y. Weiss, “On spectral clustering: analysis and an algorithm,” *Proceedings of the 14th International Conference on Neural Information Processing Systems: Natural and Synthetic,* p. 849–856, 2001.

[18] “Matplotlib: Python plotting,” Matplotlib, [Online]. Available: https://matplotlib.org/. [Accessed 8 March 2020].

[19] “Plotly: Modern Analytic Apps for the Enterprise,” Plotly, [Online]. Available: https://plot.ly/. [Accessed 8 March 2020].

[20] “seaborn: statistical data visualization,” Seaborn, [Online]. Available: https://seaborn.pydata.org/. [Accessed 8 March 2020].

[21] “pandas,” [Online]. Available: https://pandas.pydata.org/. [Accessed 8 March 2020].

[22] J. Marin, A. Biswas, F. Ofli, N. Hynes, A. Salvador, Y. Aytar, I. Weber and A. Torralba, “Recipe1M+: A Dataset for Learning Cross-Modal Embeddings for Cooking Recipes and Food Images,” *IEEE Transactions on Pattern Analysis and Machine Intelligence,* 2019.

[23] R. A. Ferdman, “Map: The Countries That Drink the Most Tea,” The Atlantic, 21 January 2014. [Online]. Available: https://www.theatlantic.com/international/archive/2014/01/map-the-countries-that-drink-the-most-tea/283231/. [Accessed 7 March 2020].

[24] M. S. Donaldson, “Nutrition and cancer: A review of the evidence for an anti-cancer diet,” *Nutrition Journal,* vol. 3, no. 19, 2004.

[25] A. Maruca, R. Catalano, D. Bagetta, F. Mesiti, F. A. Ambrosio, I. Romeo, F. Moraca, R. Rocca, F. Ortuso, A. Artese, G. Costa, S. Alcaro and A. Lupia, “The Mediterranean Diet as source of bioactive compounds with multi-targeting anti-cancer profile,” *European Journal of Medicinal Chemistry,* vol. 181, 2019.

[26] “Mapchart,” [Online]. Available: https://mapchart.net/world.html. [Accessed 7 March 2020].

[27] C. M. Lăcătușu, E. D. Grigorescu, M. Floria, A. Onofriescu and B. M. Mihai, “The Mediterranean Diet: From an Environment-Driven Food Culture to an Emerging Medical Prescription,” *International journal of environmental research and public health,* vol. 6, no. 16, 2019.

[28] “Project Jupyter,” Jupyter, [Online]. Available: https://jupyter.org/. [Accessed 8 March 2020].