

Data Science Mini Project

Cuisine & Recipe Recommendations

Group Members

Pratik Patil

MIS: 112003104

Batch: T3

Division: 2

Purna Tulsiani

MIS: 112003114

Batch: T3

Division: 2

Problem Statement

Classifying the cuisine from a given list of ingredients and recommending recipes which can be made from the available ingredients and nutrition requirements of the user.

Project Objectives

1. Classifying cuisine from the given list of ingredients.
2. Recommending Recipes from the given list of ingredients.
3. Searching Recipes and get more information about the recipe.
4. Auto-completion when user is typing and particular recipe.
5. Search similar recipes according to nutrition values.

Project Requirements

User Frontend	Made using Flask
Cuisine Classification	Logistic Regression and Voting Classifier

Recipe Recommendation	TF-IDF Vectorizer
Similar Recipes according to nutrition	Average of cosine, euclidean and hamming distance
Auto-completion feature	Done using our Dataset Recipes and Javascript
Integrating ML model with Frontend	Training the model and then dump it into a pickle file and loading the file whenever required in the frontend

Process Flow

Sr. No.	Name	Techniques used
1	<i>Data Scraping</i>	<i>Selenium and BeautifulSoup</i>
2	<i>Data Pre-processing</i>	<ul style="list-style-type: none"> • <i>Removing unnecessary characters</i> • <i>Lemmatization</i> • <i>Removing Stop Words</i> • <i>NLTK POS Tagging</i> • <i>Normalizing all nutrient values</i>
3	<i>Exploratory Data Analysis</i>	<i>Box plots, Histograms, SNS Count Plot, Bar Plot</i>
4	<i>Data Modeling</i>	<i>Logistic Regression, Voting Classifier, Bi-gram Model, TF-IDF Vectorizer</i>
5	<i>Performance Evaluation</i>	<i>Accuracy and Precision</i>

GitHub Link

[pt3002/Cuisine-and-Recipe-Recommendation \(github.com\)](https://github.com/pt3002/Cuisine-and-Recipe-Recommendation)

Data Scraping

Website 1 - [Allrecipes | Recipes, How-Tos, Videos and More](#)

- Static Website where next page navigations and clicking of buttons was not required
- Hence BeautifulSoup used for scraping
- Code - Git Repo Link - [Cuisine-and-Recipe-Recommendation/BeautifulSoup.py at master · pt3002/Cuisine-and-Recipe-Recommendation · GitHub](#)

Website 2 - [RecipeDB \(iiitd.edu.in\)](#)

- Very Dynamic website which needed user inputs for giving results and many new page navigations were required.
- Used Selenium for Scraping
- Code - Git Repo Link - [Cuisine-and-Recipe-Recommendation/Selenium.py at master · pt3002/Cuisine-and-Recipe-Recommendation · GitHub](#)

Features in Scraped Data -

- Cuisine
- Dish Name
- Ingredients
- Calories
- Carbs
- Fat
- Protein
- Recipe URL
- View Data - [Cuisine-and-Recipe-Recommendation/cuisine_final.csv at master · pt3002/Cuisine-and-Recipe-Recommendation \(github.com\)](#)

Data Pre-processing

- Removing unnecessary characters from ingredient string

```
def removing_special_characters(x):  
  
    # remove anything inside paranthesis  
    x = re.sub(r"\([^\\]+)", '', x)  
  
    # remove anything containing a digit  
    x = re.sub(r"\S*\d\S*", '', x)  
  
    # make everything lowercase  
    x = x.lower()
```

```

# lemmatize all ingredients in the string
x = lemmatize(x)

# remove non-word characters except for , and -
x = ' '.join(re.findall(r"[-, '\w]+", x))

# clean excess whitespace
x = re.sub(r"\s+", ' ', x).strip()

return x

```

- **Performing Lemmatization on Ingredients**

```

def lemmatize(x):
    for word in re.findall(r"[a-z]+", x):
        x = x.replace(word, wn1.lemmatize(word, 'n') if 's' in word[-3:] else word)
    return x

```

- **Removing Stop words from ingredients and doing NLTK POS Tagging**

```

# set of stop words to filter out
stop_words = set(stopwords.words('english'))

# not-needed terms
not_needed = ['chopped', 'all-purpose', 'divided', 'or', 'to', 'taste', 'sliced', 'a', 'needed', 'finely', 'chopped', 'mi

# define a set of POS tags to keep
keep_tags = set(['NN', 'NNS', 'NNP', 'NNPS'])

cleaned_ingredients = []

for iterator in range(len(cuisine_df.index)):
    if(iterator in cuisine_df.index):
        ingredients = ingredient_list[iterator]
        cleaned_tokens = []
        for ing in ingredients.split(", "):
            individual_cleaned_token = []
            for i in ing.split():
                tokens = word_tokenize(i.lower())
                tagged_tokens = nltk.pos_tag(tokens)
                for token, tag in tagged_tokens:
                    if tag in keep_tags and token not in stop_words and token not in not_needed:
                        individual_cleaned_token.append(token)
            if(individual_cleaned_token):
                cleaned_tokens.append(" ".join(individual_cleaned_token))
        new_ing_list = ", ".join(cleaned_tokens)
        cuisine_df.loc[iterator, 'Ingredients'] = new_ing_list

cuisine_df.to_csv('test_data.csv', index=False)

```

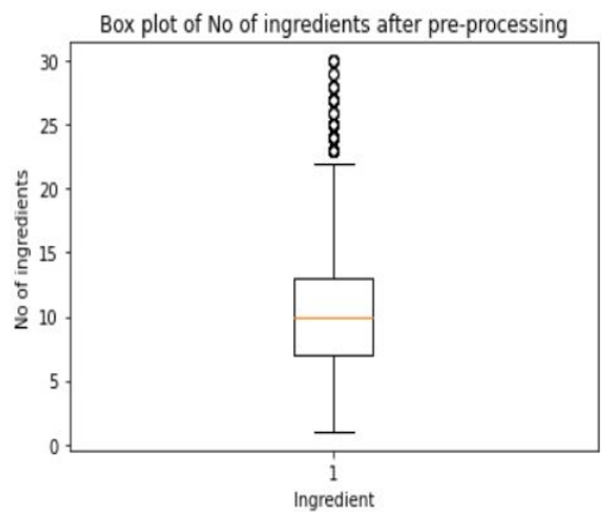
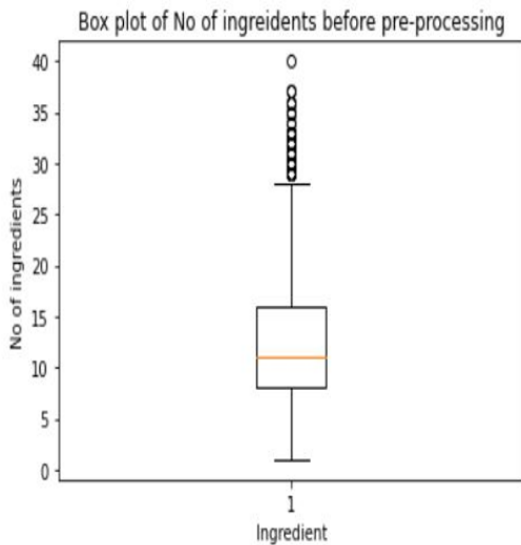
- **Normalizing Nutrients of all recipes individually according to the serving they have per gram.**

Normalized Nutrients Data Code - [Cuisine-and-Recipe-Recommendation/Nutrition_Normalized.csv at master · pt3002/Cuisine-and-Recipe-Recommendation \(github.com\)](#)

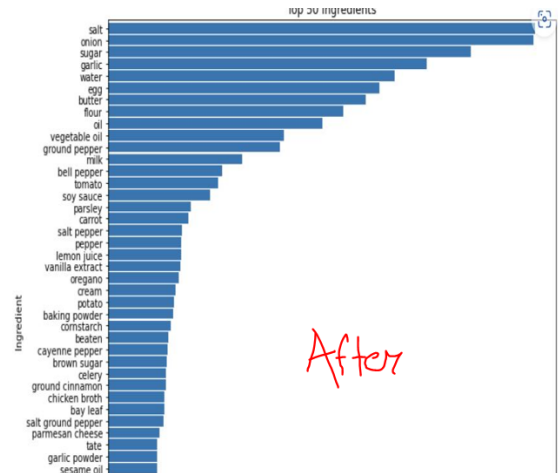
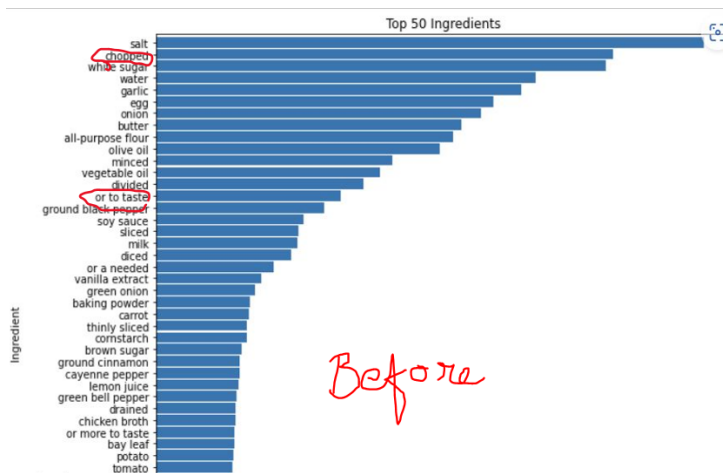
After performing all pre-processing steps:

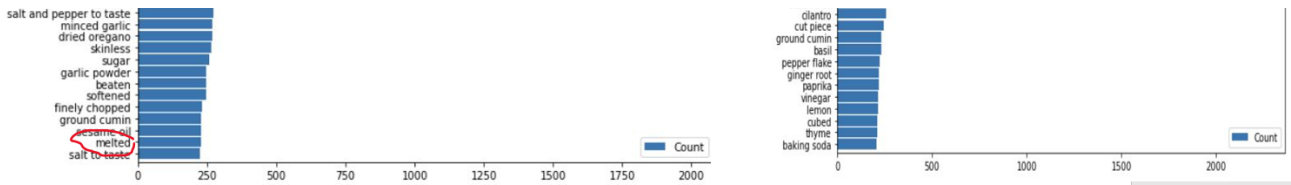
Exploratory Data Analysis

- Box plot of Number of Ingredients before and after pre-processing



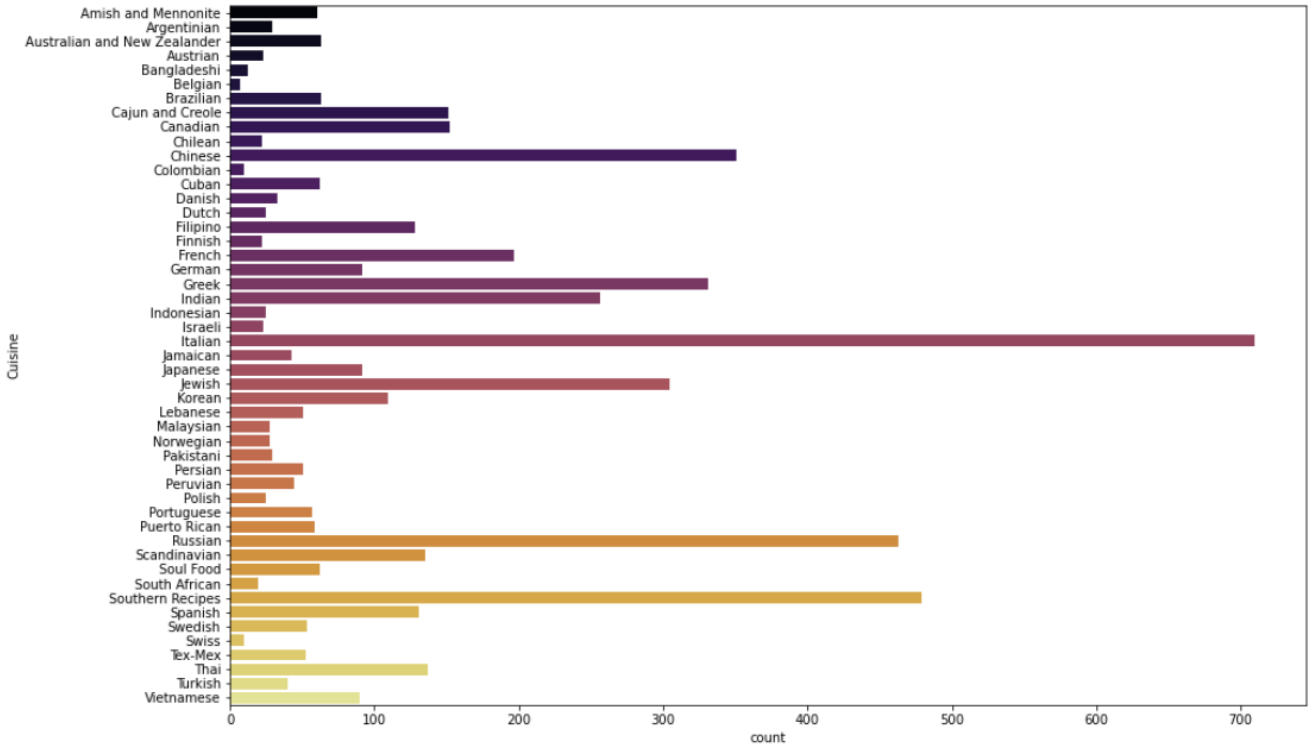
- Find out unnecessary ingredient values that are even present in the data by plotting ingredients with maximum occurrence
 - For e.g. : Chopped, according to taste, diced etc, all these words are unnecessary and need to be removed manually



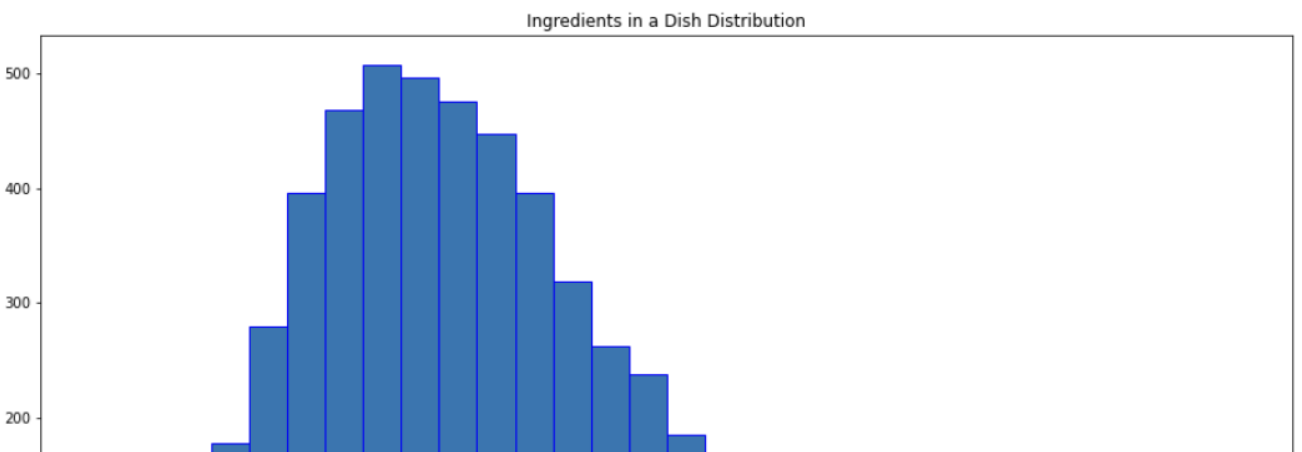


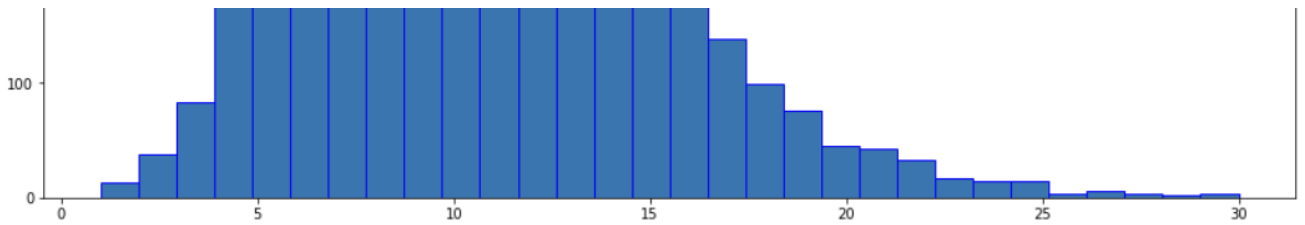
- Finding out Cuisines and Number of Recipes each cuisine has by SNS Count Plot

Cuisine Distribution

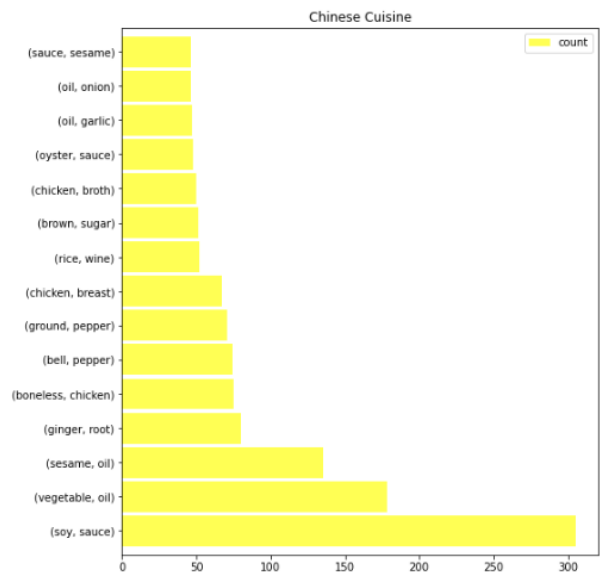
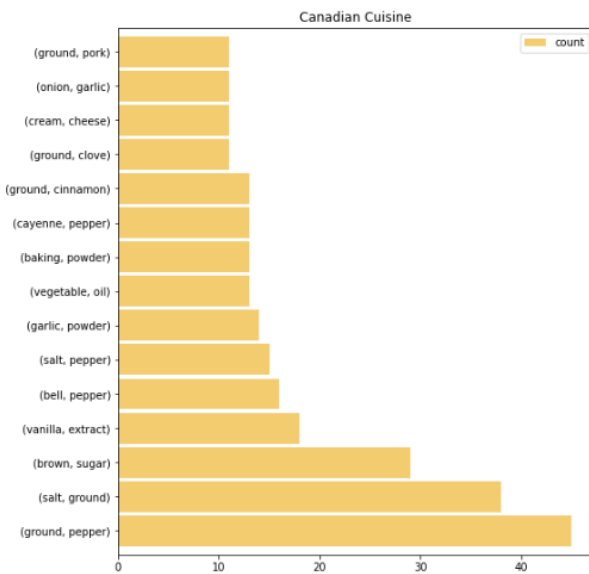
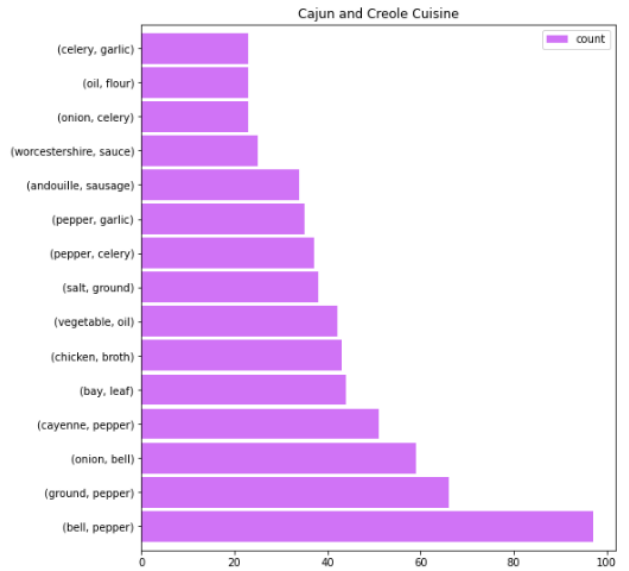
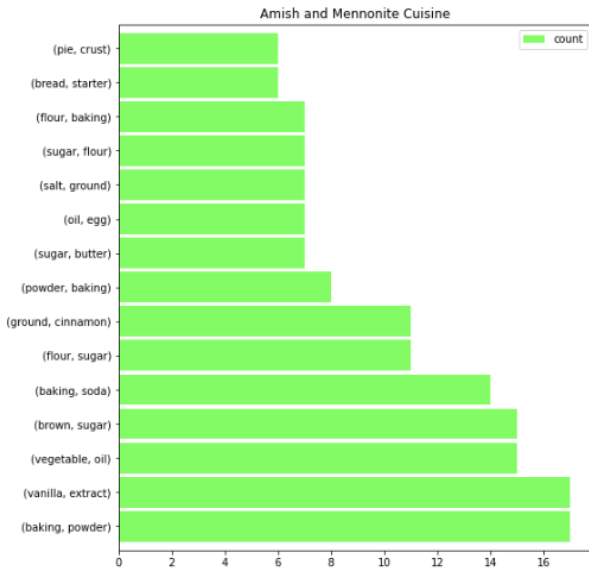


- Plotting Ingredients in a Dish Distribution





- Plotting bar plots of top 10 bi-grams of 4 example cuisines



Code of all these plots is -

Data Modeling

For Cuisine Classification -

Logistic Regression and Voting Classifier

```
In [91]: 1 from sklearn.preprocessing import LabelEncoder
2
3 encoder = LabelEncoder()
4 y_transformed = encoder.fit_transform(cuisine_df.Cuisine)
```

```
In [92]: 1 # Logistic Regression
2
3 from sklearn.model_selection import train_test_split
4 X_train, X_test, y_train, y_test = train_test_split(X_train_vectorized, y_transformed , random_state = 0)
5
6 from sklearn.linear_model import LogisticRegression
7
8 clf1 = LogisticRegression(C=10,dual=False)
9 clf1.fit(X_train , y_train)
10 clf1.score(X_test, y_test)
```

Out[92]: 0.5467255334805003

```
In [93]: 1 from sklearn.svm import SVC
2 from sklearn.ensemble import VotingClassifier
3 vclf=VotingClassifier(estimators=[('clf1',LogisticRegression(C=10,dual=False)),('clf2',SVC(C=100,gamma=1,kernel='rbf',probab
4 vclf.fit(X_train , y_train)
5 vclf.score(X_test, y_test)
```

Out[93]: 0.5695364238410596

Code - [Cuisine-and-Recipe-Recommendation/cuisinetfidf.py at master · pt3002/Cuisine-and-Recipe-Recommendation \(github.com\)](#)

Link of .pkl file - [Cuisine-and-Recipe-Recommendation/cuisine.pkl at master · pt3002/Cuisine-and-Recipe-Recommendation · GitHub](#)

For Recipe Recommendation -

TF - IDF Vectorizer

```
# TF-IDF feature extractor
tfidf = TfidfVectorizer()
tfidf.fit(df_recipes['Ingredients'])
tfidf_recipe = tfidf.transform(df_recipes['Ingredients'])

# save the tfidf model and encodings
with open(config.TFIDF_MODEL_PATH, "wb") as f:
    pickle.dump(tfidf, f)
```



```
with open(config.TFIDF_ENCODING_PATH, "wb") as f:  
    pickle.dump(tfidf_recipe, f)
```


Code - [Cuisine-and-Recipe-Recommendation/tfidfencoder.py at master · pt3002/Cuisine-and-Recipe-Recommendation \(github.com\)](#)

Link of .pkl file - [Cuisine-and-Recipe-Recommendation/tfidf.pkl at master · pt3002/Cuisine-and-Recipe-Recommendation · GitHub](#)


Front End Screenshots

Data Science Mini Project - Sem 6


Cuisine and Recipe Recommendation




Objective 1
Cuisine Prediction
From list of ingredients and we will predict the cuisine! Our dataset consists of 50+ cuisines from all over the world.
[View Feature](#)



Objective 2
Recipe Recommendation
From list of ingredients and we will predict the dishes which can be cooked, also we will provide the nutritional value of the dish.
[View Feature](#)



Objective 3
Similar Recipes Recommendation
Give us a particular recipe. And get all information from ingredients to nutritional values for top 10 most similar recipes.
[View Feature](#)



Objective 4
Search a Recipe
This objective has autocomplete feature, search any recipe and get all information related to that particular recipe.
[View Feature](#)

Designed By [Pratik Patil](#) and [Prerna Tulsiani](#)

Objective 1 - Cuisine Classification

Objective 1

Cuisine Classification from Ingredients

Enter List of Ingredients Here

tea bag ×

sugar ×

Search Cuisine

Cuisine Indian

Cuisine Southern Recipes

Designed By Pratik Patil and Prema Tulsiani

Objective 2 - Recipe Recommendation

Objective 2
 Recipe Recommendations from Ingredients

Enter List of Ingredients Here

Search Recipes

Cuisine	Recipe	Ingredients	Calories	URL
Indian	Restaurant-Style Mango Lassi	mango pulp mango slice juice plain yogurt milk ice cube	121	https://www.allrecipes.com/recipe/54319/restaurant-style-mango-lassi/
Indian	My Mango Lassi	mango plain yogurt cold milk cream confectioner sugar ground cardamom	228	https://www.allrecipes.com/recipe/264349/my-mango-lassi/
Indian	Mango Lassi-Come-Home	yogurt vanilla yogurt milk pureed mango honey mango nectar ground cardamom	207	https://www.allrecipes.com/recipe/206172/mango-lassi-come-home/
Indian	Easy Mango Lassi	plain milk yogurt milk mango sugar ground cardamom	195	https://www.allrecipes.com/recipe/202402/easy-mango-lassi/
Mexican	Mango Salsa Steak Fajitas	taco vegetable oil steak mango salsa mango juice lime jalapeno chile onion cilantro leaf	693.3845	https://cosylab.iiitd.edu.in/recipeDb/search_recipeInfo/5979

Designed By Pratik Patil and Prema Tulsiani

Objective 3 - Similar Recipes according to Nutrient Values

Objective 3
 Find Similar Recipes

Belgian Iron Cookies



Find

Cuisine: Belgian
Recipe: Pumpkin-Pompoen Confituur
Ingredients: pumpkin flesh orange lemon sugar
Calories: 605401.4504
Recipe URL: https://cosylab.iiitd.edu.in/recipeDb/search_recipeInfo/106645

Cuisine: Belgian
Recipe: Ground Beef Filling for Puff Pastry
Ingredients: onion garlic clove butter beef parsley flour worcestershire sauce
ketchup
Calories: 675401.383
Recipe URL: https://cosylab.iiitd.edu.in/recipeDb/search_recipeInfo/106651

Cuisine: Belgian
Recipe: Stoofvlees (Belgian Beef Stew)
Ingredients: beef beer peperkoek bread mustard brown sugar chocolate onion
bay leaf margarine salt pepper
Calories: 303657.5456
Recipe URL: https://cosylab.iiitd.edu.in/recipeDb/search_recipeInfo/106764

Objective 4 - Search a Recipe

Objective 4

Search a Recipe

Hong Kong Sweet and Sour Pork



Find

Cuisine: Chinese
Recipe: Hong Kong Sweet and Sour Pork
Ingredients: light soy sauce sugar potato starch sesame oil ground pepper pork
loin cut cube water vinegar ketchup sugar salt potato starch food egg beaten
potato starch peanut oil bell pepper cut chunk cayenne pepper slice pineapple
garlic onion
Calories: 588
Recipe URL: <https://www.allrecipes.com/recipe/143090/hong-kong-sweet-and-sour-pork/>

Designed By Pratik Patil and Prerna Tulsiani

Objective 5 - AutoComplete Feature

man



Herman Sourdough Starter

Moist Herman Coffee Cake

Mango Chutney

German Wiener Schnitzel

Poor Man's Pudding

Find

ALL CSV FILES -

https://drive.google.com/drive/folders/1lu6_COX1fq29Dh9e0SwWluv7vBbGiUaY?usp=sharing

Thank you!