WHAT's COOKING??

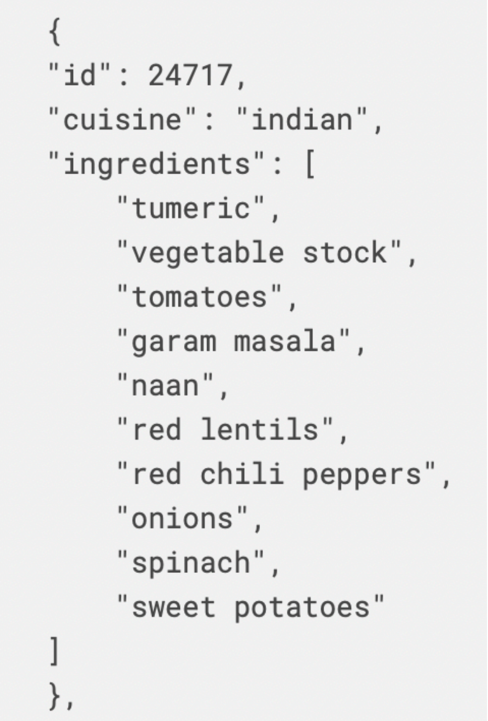
ABSTRACT

Nations or countries are frequently associated with certain foods. In a country such as the USA, we find people of various nationalities who are from all over the world. Immigrants often use food as a means of retaining their cultural identity. As people immigrate, food practices and preferences are imported and exported. And when you find yourself living in other parts of the world, having your traditional meals can be a great way to alleviate your homesickness. [1]

George Bernard Shaw said it well: "There is no love sincerer than the love of food." People from different cultural backgrounds eat different foods & the ingredients, methods of preparation and types of food eaten at different meals vary among cultures.[1]
Being a food lover myself, I have chosen the 'What's Cooking' project from the Kaggle portal.
Goal of the Project – To predict the type of cuisine of a recipe/dish on the basis of its ingredients.
One business application of this project is that it can be used by the online food delivery industry to automatically divide a new partner restaurant's menu into sub-menus based on different cuisines. Hence, expediting the process of adding a new entity into the existing system.

DATASET SUMMARY

- The data is provided to Kaggle by **Yummly**, a renowned multi-utility American website, known for providing personalized recipe recommendations.
- 2 distinct datasets in **JSON format** where individual records combine to form a list of JSON records.
  1. No. Recipes in Train Dataset – 39774
  2. No. Recipes in Test Dataset – 9944
- Can be conveniently converted to Pandas dataframes using third party libraries.
- Adjacent screenshot shows a sample record from the training dataset where each record consists of three fields / features:
  1. ***id*** – Unique identifier for a record / recipe.
  2. ***ingredients*** – JSON array containing the list of ingredients for a particular recipe.
  3. ***cuisine*** – ***Class Label*** – Based on the list of ingredients, gives the type of cuisine that record / recipe belongs to. As, it is a class label, this is not present in the test dataset.



```
{
"id": 24717,
"cuisine": "indian",
"ingredients": [
    "tumeric",
    "vegetable stock",
    "tomatoes",
    "garam masala",
    "naan",
    "red lentils",
    "red chili peppers",
    "onions",
    "spinach",
    "sweet potatoes"
]
},
```

BENCHMARKING THE THREE SOLUTIONS CHOSEN

1. Solution1 – Using XGB Classifier [2]
2. Solution2 – Using Random Forest Classifier [3]
3. Solution3 – Using Linear SVC [4]

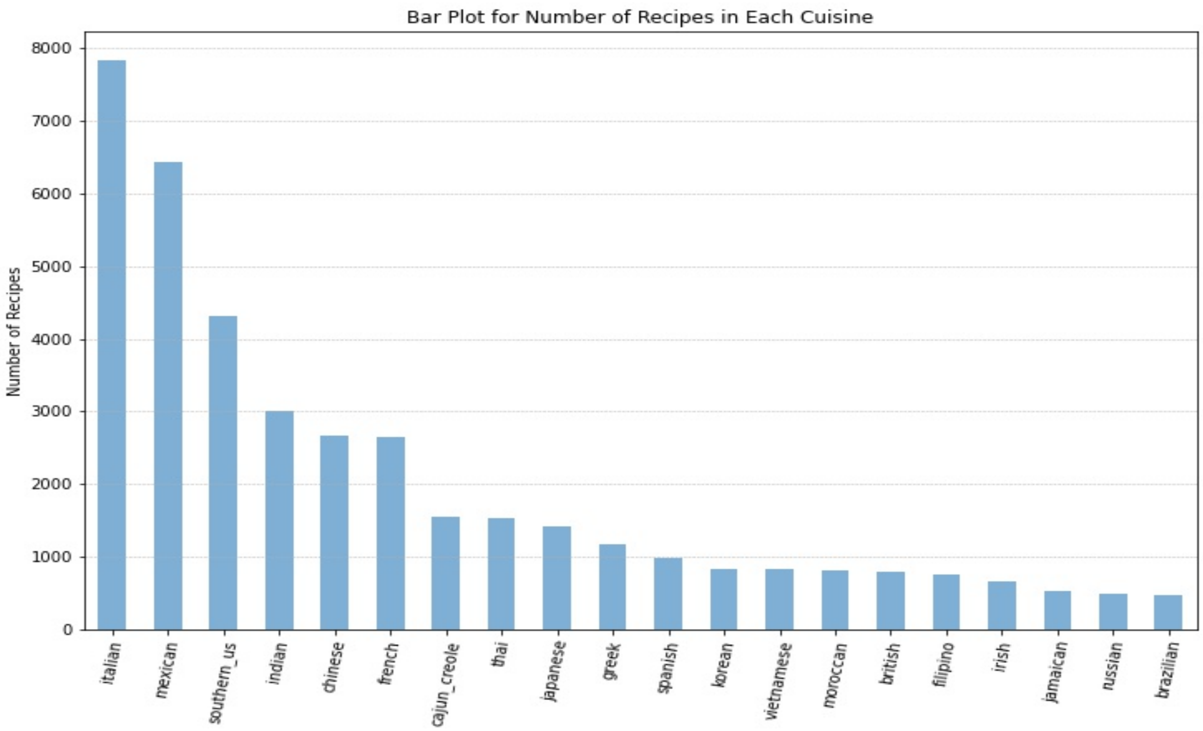
Parameters	Solution1	Solution2	Solution3
Libraries	Numpy, Pandas, Matplotlib, Seaborn, Sklearn, XGBoost	Numpy, Pandas, OS, Json, IPython, Sklearn	Numpy, Pandas, Sklearn, Time, Json, Seaborn
EDA	Performed the following activities on the training dataset: • Extracted unique ingredients for different cuisines. • Created a matrix showing the contained ingredients for a recipe from a row of all the unique ingredients. • Found top 15 (overall) ingredients used. • Identified number of recipes per each cuisine. Also, checked proportions for the same.	Performed the following activities on the training & testing datasets both: • As the given data format cannot be used in RFC, first the datasets are converted to matrices with columns those need to be every unique ingredient with a 0 or 1 if the ingredient is in a particular cuisine. • Extracted unique ingredients for different cuisines. • Using above two things, created a dataframe depicting each cuisine and ingredients present in all of its recipes.	Performed the following activities on the training dataset: • Separated out cuisines and their list of ingredients in different lists. • Transformed the list of array of ingredients to a list of strings containing the ingredients separated by a space ‘ ’. • Used TF-IDF algorithm on the **transformed corpus** to get the term frequency for each ingredient in each recipe. • Performed label encoding to normalize and encode the labels for different cuisines in order to fit LSVC model.
Model Used	**XGB Classifier** with the objective set to ‘multi:softmax’ which forces XGBoost to do multiclass classification. Also, sets the number of classes equal to unique number of cuisines in the training dataset.	**Random Forest Classifier** with number of estimators set to 10 and criterion = ‘entropy’ with the objective of computing the information gain. At last, performed cross validation to calculate the accuracy.	**Linear Support Vector Classification** with the cuisines which were label encoded in the previous step. The maximum number of iterations to be run was set to 1000.
Performance	Accuracy Achieved = 74.78%	Accuracy Achieved = 63%	Accuracy Achieved = 78.27%

Solution2 has the least accuracy and the main reason behind this is there is **no significant EDA** or data transformations before the application of model. Yes, there is a transformation to make the **data suitable for RF Classifier,** but it doesn't count as a means to increase the model's accuracy.

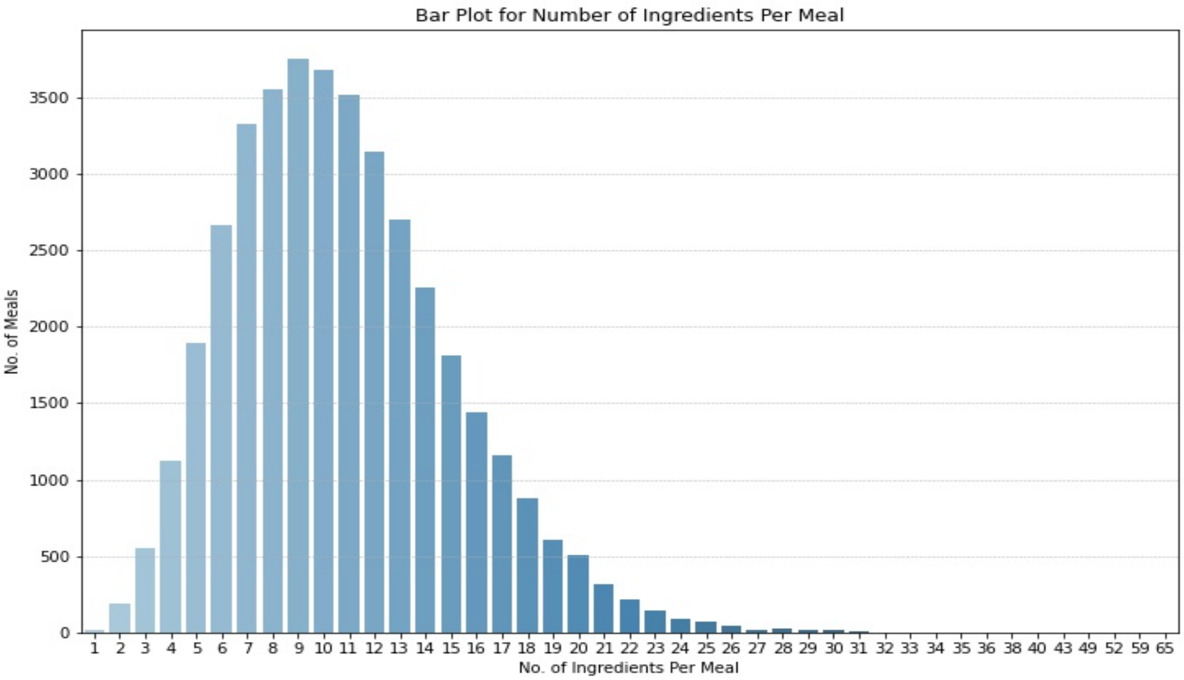
The other two solutions that are **Solution1 and Solution3 implement much more sophisticated machine learning models in the forms of XGB Classifier and Linear SVC respectively.** But this is not the sole reason for their increased accuracy as compared to the Solution2. Their EDA and other data transformations also play a critical role in increasing the accuracy of the models.

***NOTE* –** *As per the requirement for this initial report / assignment, I have not included the in-depth analysis of the three solutions chosen. For now, I have included a superficial analysis of the same and will include the detailed analysis and detailed comparisons in the final report.*

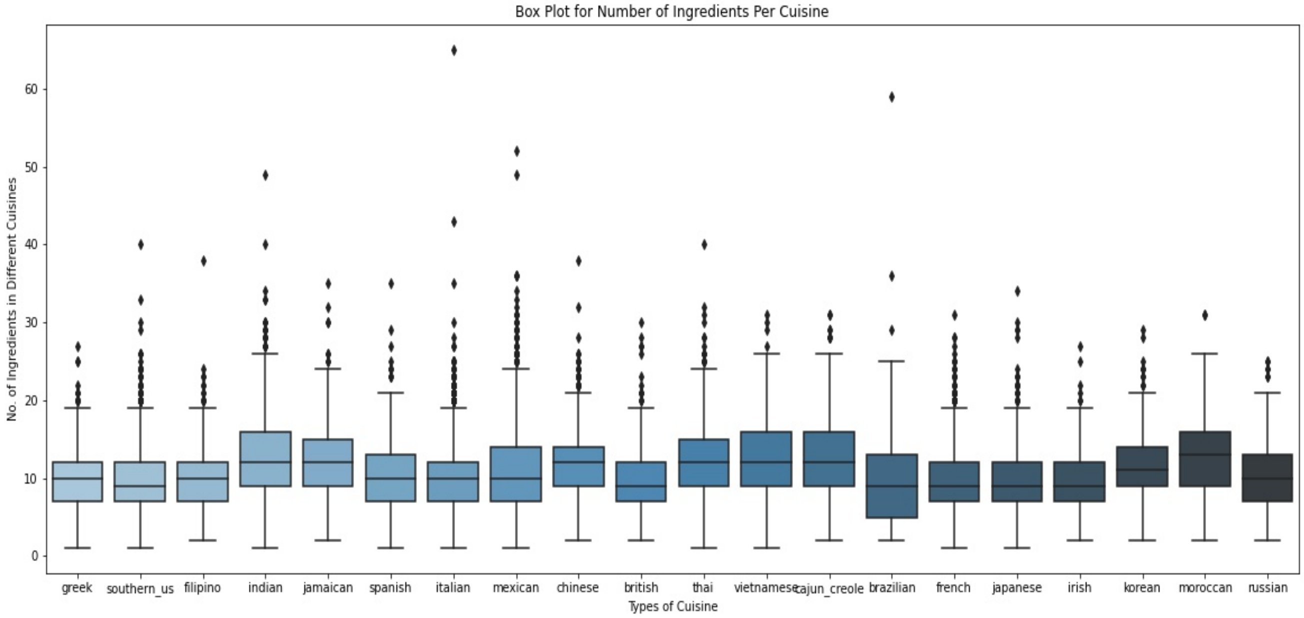
DATASET VISUALIZATIONS



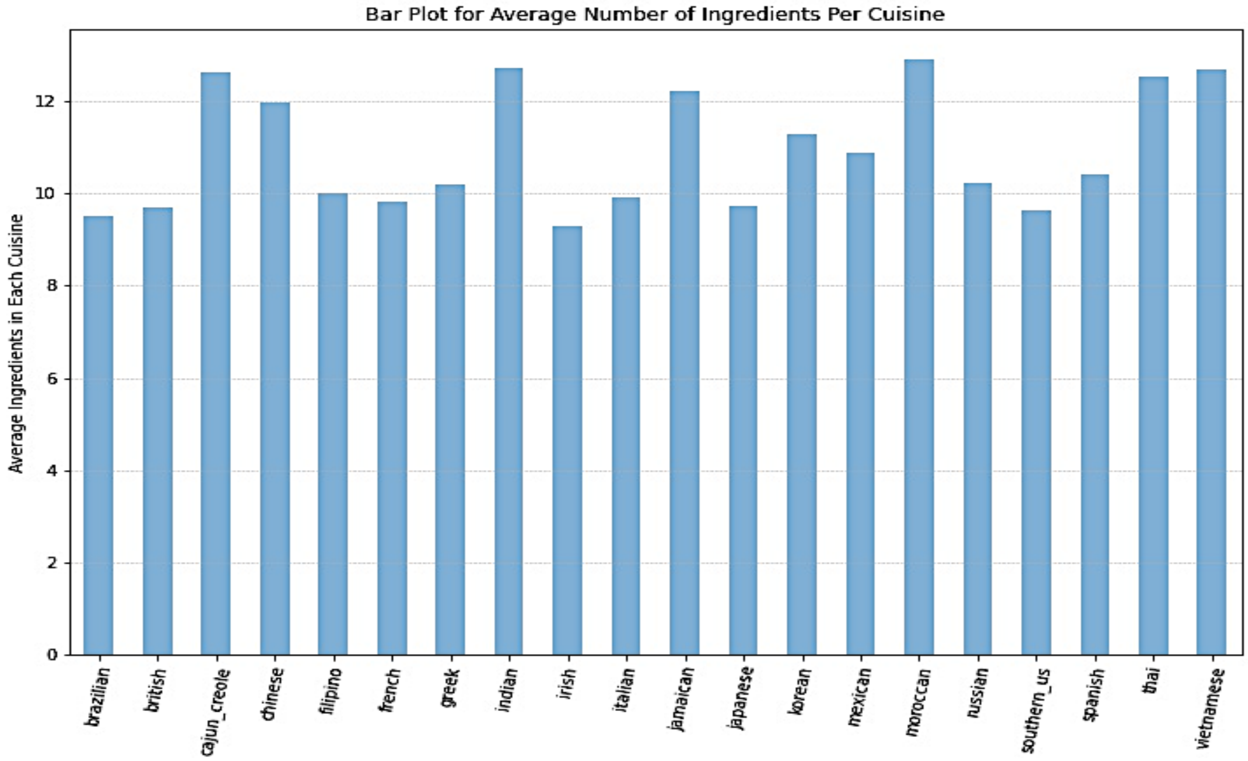
- The above plot shows the number of recipes in each of the cuisines in the training dataset.
- Out of the 39,774 recipes in the training dataset, almost 8000 recipes (maximum) belong to *'Italian'* cuisine, followed by around 6500 recipes for 'Mexican' cuisine.
- 'Jamaican', 'Russian' and 'Brazilian' are the three cuisines with the least number of datapoints with around 500 recipes each only.
- More the recipes of a cuisine in the dataset, more distinct combinations of its common ingredients. Hence, more trained the model will be.



- The above plot shows the number of meals with different number of ingredients per meal.
- Around 15,000 out of the 39,774 meals in the training dataset are cooked with 8 to 11 distinct ingredients. *It stands out to give a mean of 10 ingredients per meal.*
- However, the plot also shows that a few meals also require up to 30 distinct ingredients.



- The above box plot shows the number of different ingredients per cuisine.
- By looking at the plot, the median number of ingredients for all the cuisines may be approximated to a range of 10 to 15.
- There are also outliers (we can say those above 40 ingredients) which need to be removed from the dataset when we apply any model ourselves and carry out the analysis.



- The above bar plot shows the average number of different ingredients per cuisine.
- As per the plot the cuisines of Morocco, India and Vietnam require the highest number of ingredients to cook a meal.
- On an average each cuisine at least requires 9 ingredients for its meal to be cooked.

NOTE **–** *As per the requirement for this initial report / assignment, I have not included the in-depth description of the EDA done by myself. To plot the visualizations depicting the basic statistics for the dataset, I did EDA which included things like transforming data into suitable formats, removing null / NA values and other related things.*
For now, I have only included the visualizations and the inferences we can draw from them and will include the steps for more detailed (self-conducted) EDA and analysis in the final report.

REFERENCES

1. *https://family.jrank.org/pages/639/Food-Food-Culture.html*
2. *https://www.kaggle.com/cristianfat/what-do-we-have-for-dinner*
3. *https://www.kaggle.com/samcoh223/random-forest-classifier*
4. *https://www.kaggle.com/ranjan1701/what-scooking-using-tf-idf-linearsvc*