What's Cooking: Using Ingredients for International Cuisine Classification

[redacted] ([redacted]), Claire Windels ([redacted])

1 Introduction

Cuisine serves as a gateway to understanding the cultural and regional diversity present in our world. This project focuses on the task of categorizing cuisine based solely on the ingredients used in recipes, using the "What's Cooking?" dataset as a foundation.

This report documents the data collection, preprocessing, and visualization efforts, as well as the application of various classifiers on the dataset. Through the presentation of findings, insights, and reflections, we aim to showcase the potential of data mining in revealing patterns and connections within the culinary realm.

Please see the accompanying Jupyter Notebook for our code and full detail. This report is an overview of our thought processes, exploration, and observations.

2 Data Analysis and Visualization

Let's first delve into an analysis of the raw data provided.

2.1 Dataset Overview

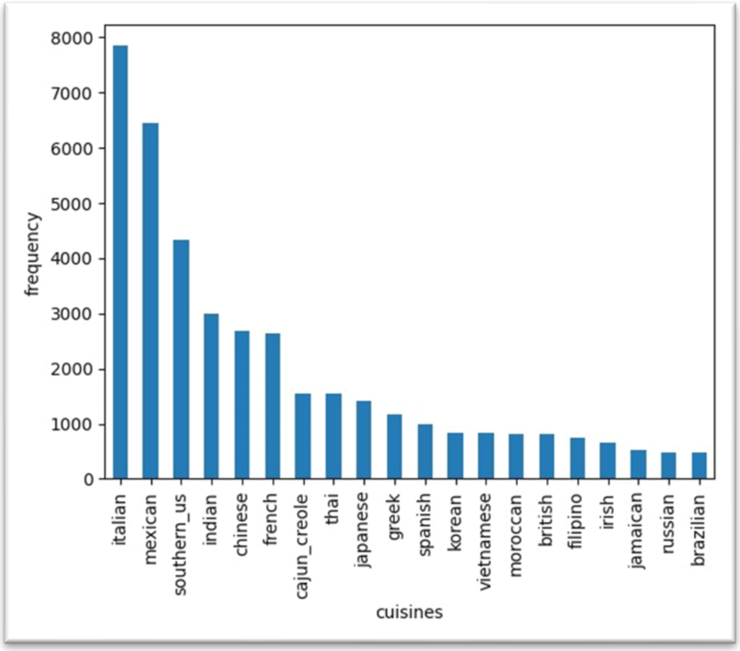
- The dataset comprises 39,774 instances.
- Each instance is characterized by three columns: 'id,' 'cuisine,' and 'ingredients.'
    - The 'id' column contains integers ranging from 0 to 49,717, serving as unique identifiers for each recipe.
    - The 'cuisine' column consists of string values representing 20 unique cuisines.
    - The 'ingredients' column contains lists of strings representing the ingredients used in each recipe.
- Notably, there are no missing or null values in the dataset, eliminating the need for imputation.

2.2 Analysis of 'id'

- The 'id' values appear to be chosen arbitrarily, lacking inherent meaning or correlation with other attributes.
- This column can be safely omitted when training and testing models as it does not contribute to predicting or understanding cuisine.
    - Excluding the 'id' column simplifies the dataset, reduces noise, and improves computational efficiency without sacrificing model accuracy.

2.3 Analysis of 'cuisine'

- The 'cuisine' column serves as the target variable in our analysis.
- As shown in the figure below, Italian, Mexican, and Southern US cuisines appear to be the most frequently represented.
    - Models may demonstrate higher accuracy for cuisines with abundant representation, benefiting from a larger sample size and more diverse training instances.
- Conversely, Jamaican, Russian, and Brazilian cuisines are less prevalent in the dataset.
    - Models predicting less frequent cuisines may face challenges due to limited data points, potentially leading to lower accuracy and poor generalization.



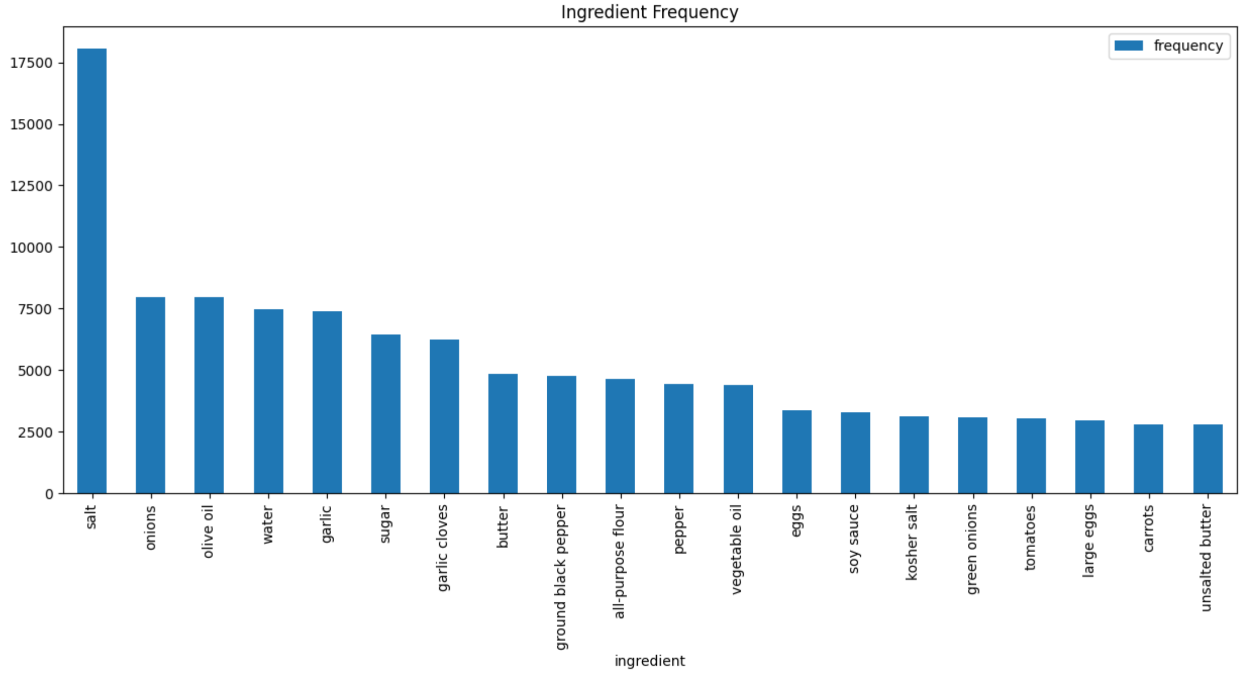
2.4 Analysis of 'ingredients'

- The predictions our models make will be based entirely on the data from this column.
- It contains 39,774 recipes, out of which 39,674 have unique ingredient lists. This indicates that there are some recipes with identical ingredient combinations.
- The length of the ingredient lists ranges from 1 to 65 ingredients.

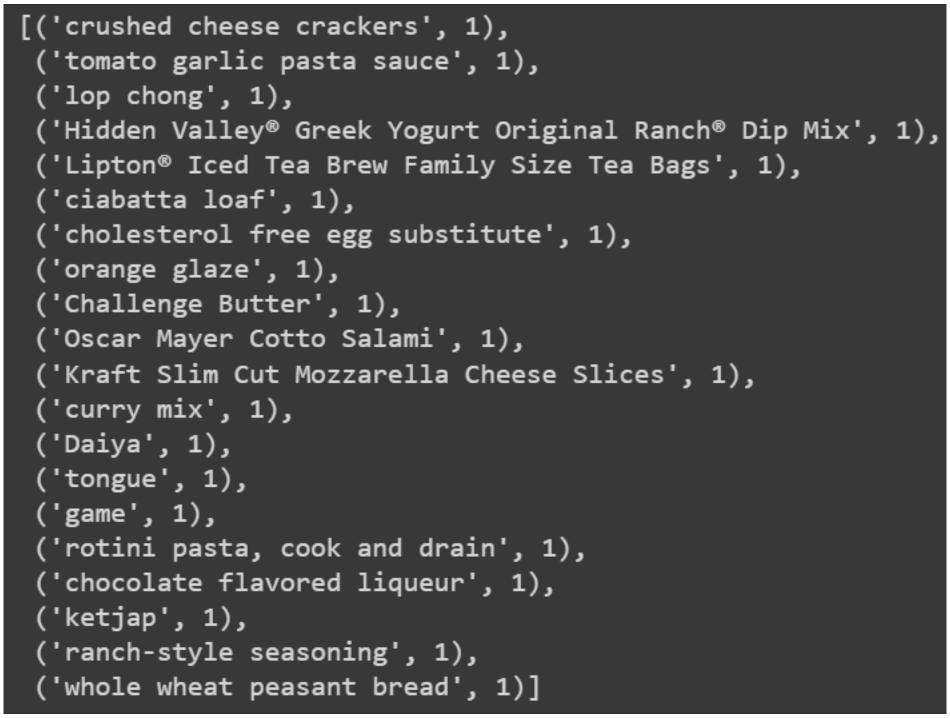
To gain further insight from the ingredient lists, it is necessary to perform text processing and analysis. This involves breaking down the ingredient lists into individual words to explore patterns, frequencies, and relationships.

To understand the diversity of ingredients in the dataset, we first identified the unique ingredients present. This was achieved by creating a set and adding each ingredient from the lists, resulting in a total of 6,714 unique ingredients.

An assessment of ingredient frequency was conducted by constructing a comprehensive list of all ingredients, including repeats. This allowed us to identify the most common ingredients present in the dataset. Unsurprisingly, we observed that ingredients such as salt, onions, and olive oil were among the most frequently used in recipes.

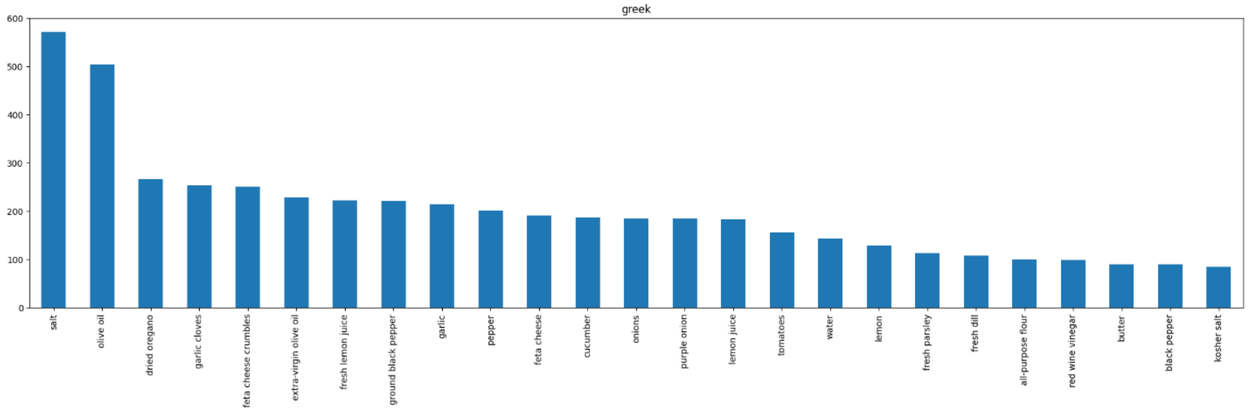


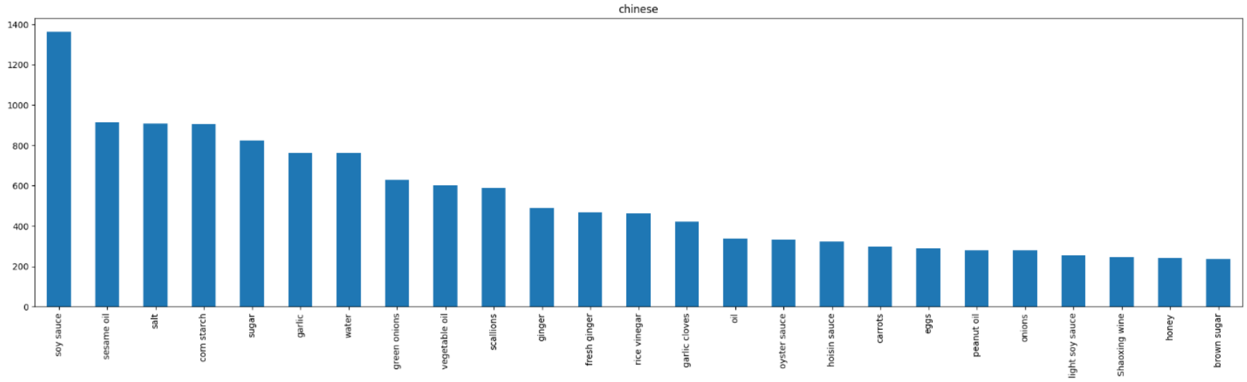
Exploring the dataset further, we identified a subset of ingredients that appeared only once. These specialized and brand-specific ingredients contribute to the uniqueness and diversity of the dataset. Noteworthy examples include 'Kraft Slim Cut Mozzarella Cheese Slices', 'Hidden Valley® Greek Yogurt Original Ranch® Dip Mix', and 'ranch-style seasoning'. We note that certain strings are capitalized and include special characters such as '®' and '-'.

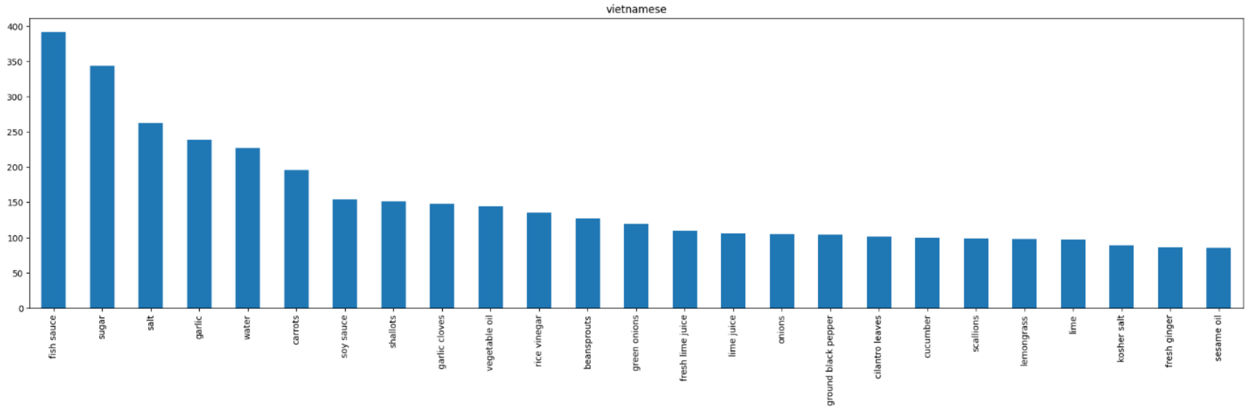


```
[('crushed cheese crackers', 1),
 ('tomato garlic pasta sauce', 1),
 ('lop chong', 1),
 ('Hidden Valley® Greek Yogurt Original Ranch® Dip Mix', 1),
 ('Lipton® Iced Tea Brew Family Size Tea Bags', 1),
 ('ciabatta loaf', 1),
 ('cholesterol free egg substitute', 1),
 ('orange glaze', 1),
 ('Challenge Butter', 1),
 ('Oscar Mayer Cotto Salami', 1),
 ('Kraft Slim Cut Mozzarella Cheese Slices', 1),
 ('curry mix', 1),
 ('Daiya', 1),
 ('tongue', 1),
 ('game', 1),
 ('rotini pasta, cook and drain', 1),
 ('chocolate flavored liqueur', 1),
 ('ketjap', 1),
 ('ranch-style seasoning', 1),
 ('whole wheat peasant bread', 1)]
```

Analyzing the ingredients specific to each cuisine, we discovered a few distinct main ingredients that occur most frequently. Salt emerged as the primary ingredient in the vast majority of cuisines, including Greek, Southern US, Filipino, Indian, Jamaican, Spanish, Italian, Mexican, British, Cajun Creole, Brazilian, French, Irish, Moroccan, and Russian. However, Chinese, Japanese, and Korean cuisines were characterized by the dominant usage of soy sauce, while Thai and Vietnamese food prominently featured fish sauce.







3 Data Transformation and Processing

3.1 Preprocessing Overview

Our preprocessing pipeline includes the following steps:

- Joining Ingredients
  - The individual ingredients within each recipe's ingredient list are concatenated into a single string, allowing for easier text processing.
- Lowercasing
  - All characters in the ingredient string are converted to lowercase, ensuring consistency in the text data.
- Removing Accents
  - Accented characters are removed from the ingredient string to normalize the text and avoid potential inconsistencies.
- Removing Non-Alphabetical Characters
  - Non-alphabetical characters, such as punctuation marks and special symbols, are removed from the ingredient string. This step helps to streamline the text data and remove noise that may interfere with subsequent analysis.
- Removing Common Units
  - Common units of measurement, such as ounces, lbs, and kg, are removed from the ingredient string. These units do not contribute to the essence of the ingredient itself and can be considered irrelevant for the purposes of analysis.
- Lemmatization with NLTK
  - Lemmatization is applied to the processed ingredient string using the Natural Language Toolkit (NLTK). Lemmatization reduces words to their base or root form, allowing for more accurate analysis by treating different forms of the same word as a single entity. This step helps to address variations and ensure consistency in the ingredient representation.

By following this preprocessing pipeline, the ingredient data is transformed into a more standardized and analytically useful format, facilitating subsequent modelling and classification tasks.

3.2 Processing Attempts

3.2.1 Bigrams

Implementation: Bigrams involve analyzing pairs of consecutive words in the ingredient text, thereby capturing contextual information in the data. We incorporated bigrams into the preprocessing function using the ngrams() function to generate the pairs of words and then join them with an underscore.

Impact on Accuracy: Surprisingly, the inclusion of bigrams in the preprocessing pipeline resulted in lower model accuracy. One possible reason is that bigrams might introduce noise or irrelevant information by considering word pairs that do not significantly contribute to the predictive power of the models.

Additionally, the increased dimensionality introduced by bigrams might lead to overfitting, where the models memorize the training data rather than generalizing well to unseen data.

3.2.2 Spelling Correction with TextBlob

Implementation: We attempted to improve the accuracy of the models by employing spelling correction using the TextBlob library. This process aims to rectify misspelled words in the ingredient text.

Impact on Efficiency: Unfortunately, the use of TextBlob for spelling correction proved to be highly time-consuming and inefficient. It was unable to finish running within a 3-hour time frame. This indicates that this approach is not scalable and might become a computational bottleneck when dealing with larger datasets.

(Failed) Alternative Solution:

As a replacement for the time-intensive spelling correction with TextBlob, we sought a more time-efficient spell-checking solution. We attempted to run a spell check on the list of 6,714 unique ingredients so we could manually correct the most crucial misspelled ingredients. Unfortunately, this alternative approach also proved to be time consuming, having not finished executing after an hour.

3.3 TF-IDF Vectorizing

We use the TfidfVectorizer function from scikit-learn to convert the ingredient text data into numerical feature vectors using the Term Frequency-Inverse Document Frequency (TF-IDF) scheme. Stop words, common words with limited informational value such as ‘and’, ‘a’ and ‘an’, were removed during vectorization to improve feature quality. Furthermore, label encoding was applied using the LabelEncoder function to transform the categorical cuisine classes into numerical representations. The resulting feature matrix (X) contains the TF-IDF representations of the ingredients, while the target array (y) contains the numerical cuisine labels.

4 Model Selection

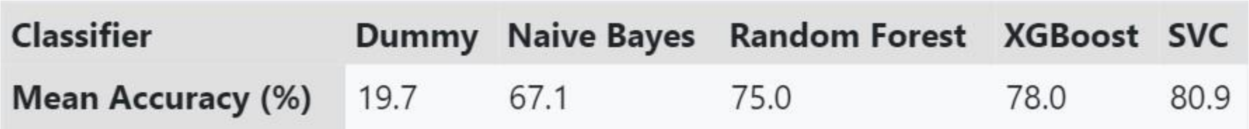
We trained five different commonly used classifiers with our data and evaluated their performances using 5-fold cross-validation. The classifiers used, in the ascending order of accuracy scores, are as follows:

1. Dummy Classifier
    - This classifier serves as a baseline model, making predictions based on the most-frequent class label in the dataset.
    - Given its simplistic nature, it is expected to have the lowest accuracy compared to other more sophisticated models.
2. Multinomial Naïve Bayes
    - Naïve Bayes is a probabilistic classifier based on Bayes' theorem, commonly used for text classification tasks.
    - It assumes that features (ingredients) are conditionally independent given the class (cuisine).

  - The accuracy is moderate, it could be capturing certain ingredient-class associations well but may be oversimplifying some more complex interactions.

3. Random Forest
   - Random Forest is an ensemble learning method that constructs multiple decision trees and combines their predictions to improve accuracy and reduce overfitting.
   - It is capable of handling non-linear relationships and may yield better performance than Naïve Bayes by capturing complex interactions among ingredients.
4. XGBoost
   - XGBoost is an advanced gradient boosting algorithm known for its efficiency and accuracy.
   - It excels in handling large datasets, capturing intricate feature interactions, and performing well in text classification tasks like this one.
   - Interestingly, it performs worse than the Support Vector Classifier. It would likely perform better on a larger dataset, or perhaps with more intensive ingredient preprocessing.
5. SVC (Support Vector Classifier)
   - SVC is a powerful linear and non-linear classifier that aims to find the best hyperplane to separate classes in high-dimensional space.
   - SVC produces the highest accuracy in our case. We suspect this is because we are training and testing it on a medium-sized dataset, which it handles well.

The following is a table that summarizes the computed model accuracies.



Classifier	Dummy	Naive Bayes	Random Forest	XGBoost	SVC
Mean Accuracy (%)	19.7	67.1	75.0	78.0	80.9

5 Improving our Model

In this phase of our analysis, we focused on optimizing the hyperparameters for the Support Vector Classifier (SVC) using GridSearchCV. We defined a hyperparameter grid encompassing various values for C, gamma, and the kernel type (linear and rbf). By exhaustively evaluating the combinations within this grid through 5-fold cross-validation, we identified that the hyperparameters that yield the highest accuracy for our dataset are C=10, gamma=1, and kernel=rbf. Ultimately, the hyperparameter selection did not increase our accuracy significantly. The final mean accuracy was still 80.9%.

6 Conclusion

In conclusion, this project explored the task of categorizing cuisines based on ingredient data using various data mining techniques. Through data analysis and visualization, we gained insights into the dataset's characteristics, including cuisine distribution and ingredient frequencies. We employed preprocessing techniques such as lowercase conversion, accent removal, and lemmatization to transform the ingredient text into a standardized format for modelling.

Among the classifiers tested, the Support Vector Classifier (SVC) demonstrated the highest accuracy for our medium-sized dataset. This can be attributed to SVC's ability to handle high-dimensional data and find optimal hyperplanes for classification. However, further consideration should be given to improving the text preprocessing stage. Exploring alternative techniques for spelling correction and handling brand-specific ingredients could enhance the quality of the ingredient data and potentially improve model accuracy.

To improve model accuracy, it would be worthwhile to investigate advanced techniques such as neural networks or ensemble models. Additionally, incorporating domain-specific knowledge, such as culinary and food-related data and resources, could enhance the performance of the classifiers. Finally, expanding the dataset by collecting more recipes from a broader range of cuisines would provide a richer and more diverse training set, allowing models to capture more nuanced relationships between ingredients and cuisines.

In summary, this project demonstrates the potential of data mining in the culinary domain by successfully categorizing cuisines based on ingredient data. Through further improvements in text preprocessing and model selection, along with the incorporation of domain expertise and expansion of the dataset, more accurate and robust classifiers can be developed to better understand and explore the intricate relationships between cuisines and ingredients.