Cuisine Predictor: Classifying Recipes through their Ingredients

Aarushi Arora
Natural Language Processing
Occidental College

Layal Bata
Natural Language Processing
Occidental College

I. Introduction

Every culture's cuisine tends to have a number of techniques or ingredients that are unique to that cuisine. With that comes recipes, the documentation of preparing that food, including the ingredients, techniques, and steps special to the dish. For example, words like garlic, basmati rice, tomato, stir, and coconut milk are common in recipes of the Indian subcontinent. Because there is some repetitiveness in the ingredients and processes used within a cuisine, classifying recipes by the ingredient words into their cuisine is an interesting question. We can use natural language processing techniques to take a closer look at the patterns of ingredients within cuisines to predict the cuisine of

II. Related work

Because of how much recipe data is available, there has been a lot of previous research looking at classifying recipes or cuisines based on different factors. T Sharma et. al [1] used Text Classification methods to classify recipes through their processes and techniques (stir, deep fry), utensils, as well as ingredients. This gave insight into how all aspects of cooking that are included in the recipe could be used for classification, and how various models (Naive Bayes, Logistic Regression, Support Vector Machine, etc.) would perform with this data. The data this work used however was much more vast and had a lot of non-specific terminology, so we wanted to use a smaller dataset focused on ingredients to evaluate how well a recipe could be evaluated using the food items alone.

Other work has been done with recipe data in which researchers use the text in recipes and their organization method to recommend other recipes. They used cooking processes to classify by type of recipe and then looked at the ingredients, and found that different regions had more commonly used cooking processes. We then want to examine to what extent this could be said for ingredients how well it would be classified.

Looking past recipes, a lot of research has been done on NLP classification methods, where we looked at classification done on Movie scripts to figure out their genre as it has a similar structure to recipes [3] as well as music lyrics to classify by genre.[4] Recipes are to scripts or lyrics as cuisine is to the genre (cuisines are essentially the 'genre' of food). Although the corpus here is movie scripts as opposed to recipes, the process of building their model looks very similar to what we are anticipating we need to do to answer our research question. We're anticipating our feature extraction to also have similar questions; where the researchers here wonder whether genre is plot dialogue driven or scene driven, we have to question whether a recipe's categorization is technique driven or ingredient driven, and to what extent these features interact.We also examined work done with general food classification into their Allergy or Tolerance group. Although this paper doesn't classify recipes, it connects with our intended project because of how it uses text classification: classifying the foods into a certain category, which is the same structure as ours (classifying recipe to cuisine, which helped us understand the process we could take to evaluating our recipes and results. [5]

III. methodology

A. Data Collection

We found our data set, entitled "Recipe Ingredient Data Set" on the public website Kaggle. This dataset, kept in a .json file, contains a pre-made training and testing set that includes the recipe id, the cuisine, and all the ingredients for that recipe. Here is an example of one recipe node in the training set:

{
"id": 24717,
"cuisine": "indian",
"ingredients": [
"tumeric",
"vegetable stock",
"tomatoes",
"garam masala",
"naan",

```
        "red lentils",
        "red chili peppers",
        "onions",
        "spinach",
        "sweet potatoes"
    ]
    },
```

The testing data set is structured the same except the cuisine type is removed so it can be predicted.

B. Methods

We wrote our code in Python using the program PyCharm with a Logistic Regression. We began by importing a number of libraries available. Most importantly, we imported nltk in order to take care of our basic natural language processing needs, as well as pandas to enable some statistical analysis and use of .json files. We then imported some modeling tools from the sklearn library, like feature extraction and pipeline.

We then imported our files for our train and test sets, and started them at the beginning of the files. Following this, we went to get some basic information about our training data, such as the number of ingredients, the number of cuisines. and each ingredient. We then created an ingredient vocabulary. With all this set up, we could train our model. Our classifier uses the Logistic Regression model pipeline function of the sklearn library, equipped to classify with vectorization using TF-IDF. After that, we wrote a definition to turn our data into a string, calling it on our list of ingredients in the training set.

From here, we fit and scored our model by cuisine type, and then called on predict to go through and predict the cuisine of each list of ingredients in the training set. We then use the metrics from the sklearn library to get some tables of accuracy to analyze. At this point, we planned on finding helpful features and picking ones to help our Logistic Regression model be more accurate, but we had to cut this in order to have working code.

Finally, we turned to our testing data, made it a string, and predicted the cuisine of each list of ingredients. We took these predictions and imported them into a .csv file for evaluation.

III. Results and analysis

Overall, on the training data, our precision, recall, anf f1 scores vary widely across cuisines. Looking to Figure 1, we can see that while some cuisines, like US Southern food, can be categorized as such with very high scores across the board, our model struggled to identify other cuisines consistently, like Chinese food with a precision score of 0.51 and recall 0.37.

	precision	recall	f1-score	support
cajun_creole	0.68	0.41	0.51	467
chinese	0.51	0.37	0.43	804
moroccan	0.68	0.65	0.66	1546
mexican	0.75	0.83	0.79	2673
brazilian	0.65	0.44	0.52	755
italian	0.64	0.44	0.53	2646
greek	0.73	0.57	0.64	1175
russian	0.81	0.91	0.86	3003
indian	0.71	0.24	0.36	667
irish	0.73	0.89	0.80	7838
french	0.71	0.63	0.67	526
japanese	0.75	0.69	0.72	1423
jamaican	0.77	0.71	0.74	830
southern_us	0.87	0.91	0.89	6438
spanish	0.78	0.71	0.74	821
filipino	0.68	0.36	0.47	489
korean	0.63	0.75	0.69	4320
thai	0.68	0.38	0.49	989
vietnamese	0.71	0.76	0.73	1539
british	0.71	0.42	0.53	825

Fig. 1. Comparison of precision, recall, and f1 scores of our training set across cuisines

Across cuisines, we still have an average accuracy of about 74%, as shown in Figure 2.

accuracy			0.74	39774
macro avg	0.71	0.60	0.64	39774
weighted avg	0.73	0.74	0.73	39774

Fig. 2. Average accuracy of full model on training data

With our model predicting the cuisine with just under 75% accuracy, we felt comfortable moving to the test set. Because there is no quantitative way to look at our predictions and discern their accuracy as we did with our training data, we looked through the data to see if we could find any errors with our own knowledge. We put the IDs into a random number generator and chose 10 predictions to inspect. While there were some examples in our final prediction file that did not seem accurate, like the predicting of a dish with salmon in it as Indian when most of the recipes we saw for the cuisine in our set were vegetarian, it did not seem to be significantly less accurate on our testing set than our training set, as 7/10 of the predictions we looked at seemed correct. Although this test set is small, it seems to be on par with what we were expecting based on the training data, so we feel confident in stopping there for the current inspection.

Overall, documenting the processes in recipes is a practice in cultural preservation. Because of how much data about food and recipes there are, it can be difficult to categorize and organize, and using natural language

processing techniques to categorize this content seems very promising going forward This could allow better techniques for preserving these recipes, because with the number of recipes all over the internet, it can be hard to find and keep track of, so being able to identify and tag what cuisine a recipe is part of can have significant implications for recipe collection as well as cultural preservation across time. We also hope this technique can also be used for other NLP applications such as generating new and unique recipes for a cuisine or recommending recipes in the future.

References

[1] Sharma, T., Upadhyay, U., & Bagler, G. (2020, April). Classification of Cuisines from Sequentially Structured Recipes. In *2020 IEEE 36th International Conference on Data Engineering Workshops (ICDEW)* (pp. 105-108). IEEE.

[2] Teng, C. Y., Lin, Y. R., & Adamic, L. A. (2012, June). Recipe recommendation using ingredient networks. In *Proceedings of the 4th Annual ACM Web Science Conference* (pp. 298-307).

[3] Blackstock, A., & Spitz, M. (2008). Classifying movie scripts by genre with a MEMM using NLP-Based features. *Citeseer*. https://nlp.stanford.edu/courses/cs224n/2008/reports/06.pdf

[4] Zheng, E., Moh, M., & Moh, T. S. (2017, January). Music genre classification: A n-gram based musicological approach. In 2017 IEEE 7th International Advance Computing Conference (IACC) (pp. 671-677).

[5] Campese, S., & Pozza, D. (2020, September). Food Classification for Inflammation Recognition Through Ingredient Label Analysis: A Real NLP Case Study. In *Proceedings of SAI Intelligent Systems Conference* (pp. 172-181). Springer, Cham.