Mia Jeffries, Trisha Bhima, Talisa Pham Quang
Professor Meiqing Zhang
COMP331 - Natural Language Processing
10 December 2025

Recipe Classification with Machine Learning and Large Language Models

i. Introduction

The ability to classify recipes into meaningful categories is an important task in culinary data analysis, especially when it comes to organization, search, and recommendation. This project investigates the task of recipe classification in order to explore how both traditional machine learning techniques and modern natural language processing methods can be applied to recipe text data in order to accurately predict generalized recipe categories.

ii. Methodology

The data used for this project was sourced from a Kaggle user who has webscraped data from Food.com into a single dataset called “Food.com - Recipe and Reviews” (Karltonkxb). The starting data file was recipe.csv, which contained over 500,000 recipes across 312 different categories. For each recipe, the following information was recorded: name, author, cook time, prep time, total time, date published, description, images, recipe category, keywords, recipe ingredient quantities, recipe ingredient parts, rating, review count, various nutritional information, and recipe instructions.

First, the raw data was processed and saved into a new file that could be better used with machine learning models. Many of the starting variables were dropped, leaving the variables Name, RecipeCategory, RecipeIngredientParts, and RecipeInstructions. These variables were kept as they contain the core semantic information that is important to recipe classification. To further streamline the dataset, each of the remaining variables was individually cleaned. The name of each recipe was lowercased and stripped for consistency. The originally assigned recipe category of each recipe – which consisted of 195 unique values – was mapped to one of nine target generalized keywords: beverages, dessert, breads, soups, main dish, sides, sauces, snacks, and breakfast. Because the initial categories were relatively narrow and fragmented, the dataset was relatively noisy and imbalanced, and the choice to generalize each recipe’s category helped

the dataset become more structured and learnable for classification tasks. Additionally, many of the starting categories were cuisine labels that encapsulated recipes across a variety of food "types" and were difficult to place within the target labels, thus they were separated out for potential future feature engineering. Finally, both the RecipeIngredientParts and RecipeInstructions variables were lowercased and stripped of elements to be in plain text form. The resulting dataset – processed_data.csv – was ready for model training and contained the variables Name, Recipe Category, Ingredients, and Instructions.

With this finalized dataset, a baseline traditional machine learning algorithm was fit using a Random Forest classifier. The goal of this was to see how well a bag-of-words representation and tree-based classier would be able to classify a bulk amount of recipes into consolidated target variables. To fit the model, a stratified train/test split was used to help adjust for potential data imbalances and was split into 80% and 20% respectively. The model was trained on TF-IDF vector representations of the recipes that combined the aforementioned ingredients and instructions variables. For this, the maximum feature count was set to 5000, unigrams and bigrams were extracted, and English stop words were removed to reduce noise. The Random Forest itself consisted of 150 decision trees and no maximum depth in hopes that trees could grow fully and completely to best capture feature complexity.

As a secondary classification model, a pretrained DistilBERT model was fine-tuned on the processed data. Doing so allowed for comparison between recipe classification techniques. Ingredients and instruction variables from the processed dataset were combined into a single textual feature – Recipe – and the original ingredients and instructions columns were dropped for model training training. Any missing or null values in Recipe were replaced with empty strings, and underrepresented categories (e.g. soups and other) were merged into one of the others to improve class balance. After these merges, the final dataset for the DistilBERT model consisted of eight categories: Beverages, Breads, Breakfast, Dessert, Main Dish, Sauces, Sides, and Snacks. Finally, labels were encoded from strings to numeric IDs using LabelEncoder, and a label map was saved for future reference. To perform model training, the dataset was split into stratified training, validation, and test sets, which were made 70%, 10%, and 20% respectively. DistilBertTokenizerFast was used to tokenize the Recipe text; sequences were padded or truncated to a maximum of 256 tokens in order to standardize input length for the model. Training was carried out using a Trainer API with the following hyperparameters: a learning rate of 2e-5, a train batch size of 16, an evaluation batch size of 16, and a weight decay of 0.01. The

model was trained for 3 epochs over the training split, and the trained model was evaluated on the validation and test sets.

iii. Results

The baseline Random Forest achieved an overall accuracy of 0.6763 and strong performance on beverages, dessert, breads, and main dishes, which is likely due to their unique features or large dataset size. Smaller or more nuanced categories – such as soups and snacks – resulted with lower metrics. A classification report of the Random Forest's performance was recorded and saved to its own file, highlighting precision, recall, f1-score, and support over all labels and accuracy overall. These results were able to provide a clear benchmark that future model iterations could be compared against. Overall, it seems that the Random Forest was capable of capturing basic linguistic features and patterns, though a more sophisticated model would likely capture deeper semantics more thoroughly.

The fine-tuned DistilBERT model achieved a test accuracy of 0.7416 after carrying out the recipe classification task. This reflects a clear improvement over the 0.6763 accuracy achieved by the Random Forest baseline, indicating that contextualized transformer representations are better suited to capture detailed semantic patterns of recipe ingredient and instruction text compared to the TF-IDF features used in Random Forest.

Below is a table with category-specific results of the DistilBERT model:

Category	**Precision**	**Recall**	**F-1 Score**
Beverages	0.8863	0.8654	0.8757
Breads	0.8039	0.8446	0.8237
Breakfast	0.7337	0.6880	0.7101
Dessert	0.8669	0.9201	0.8927
Main Dish	0.7534	0.8062	0.7789
Sauces	0.6160	0.5991	0.6074

Sides	0.6266	0.6663	0.6458
Snacks	0.5430	0.3331	0.4129
Average	0.7287	0.7153	0.7184

Table 1: Category-specific DistilBERT results

These metrics show that DistilBERT performed consistently across categories, with above average results in the Beverages, Breads, Dessert, and Main Dish classes as well as meaningful improvements over the baseline in most areas. Overall, the final results demonstrate that DistilBERT provides good performance on the task, an improvement from the Random Forest that effectively uses contextual information in its classification.

Overall, the Random Forest was able to capture some strong lexical patterns, especially for categories with distinctive vocabularies such as Beverages and Desserts. However, TF-IDF cannot capture word order, compositional relationships between ingredients, or context-dependent meanings of tokens. On the other hand, DistilBERT learns contextual representations of each subword token conditioned on the entire recipe text, allowing the model to pick up subtler cues such as phrases describing cooking methods or ingredient combinations that may be indicative of specific recipe categories.

iv. Discussion

There are some limitations within this project that are important to acknowledge. First, the label space is derived from manually mapping a large number of original Food.com categories into a smaller set of generalized labels, and then later merging underrepresented classes such as Soups and Other. While this makes the classification problem more tractable and improves class balance, it also introduces some label noise and conceptual ambiguity, and the distinction between different recipe categories was not always clear-cut. Thus, there were some potential category overlap issues amongst the mapped dataset. Furthermore, there were some pre-existing category issues and discrepancies in the original dataset which may have complicated the model's ability to learn clean decision boundaries.

Additionally, we only relied on textual information when training despite using both the Ingredients and Instructions variables. The original dataset includes potentially useful data such as nutritional information, rating, and review count that could help disambiguate recipes that are

textually similar yet intended for different contexts. For example, snack recipes might have different typical serving sizes or be described differently in user reviews when compared to full meals. In the future, we could add to the feature space by incorporating additional fields from the original Food.com dataset. Adding cuisine labels, nutrition profiles, or even aggregated review text could give the model more context about how a recipe is used and perceived, potentially improving classification performance. It would also be interesting to explore a multi-label version of the dataset; some recipes may naturally belong to multiple categories — like a breakfast snack or a dessert beverage — and a multi-label setup could capture this more faithfully.

v. Conclusion

All in all, this project investigated the task of recipe classification using both a traditional Random Forest as a baseline and a modern transformer-based DistilBERT model. Beginning with a large web-scraped Food.com dataset, information was cleaned and consolidated into a manageable set of labels, and a processed dataset that combined ingredients and instructions for each recipe was constructed. The Random Forest baseline – trained on TF-IDF features – provided a strong starting point with an accuracy of 0.6763 and highlighted the categories that may be easier or more difficult to classify using simple lexical features.

Building on this, we fine-tuned a DistilBERT sequence classification model for the recipe classification task, using a manipulated version of the processed data as text input and targets. The DistilBERT model achieved an accuracy of 0.7568 on the test set, an improved performance across most categories, particularly for large and semantically distinctive classes such as Dessert and Main Dish. While some ambiguity remains for overlapping categories and noisy labelling, this work shows that applying modern NLP techniques to structured recipe data can yield meaningful improvements over traditional methods and opens up opportunities for further exploration.

vi. Code Availability

All code for this project can be found on Github at this link: https://github.com/miazjeffries/Final-Project---ML-Recipe-Classification

Works Cited

Karltonkxb. “Food Recipe - Association Rules.” *Kaggle*, Kaggle, 20 Jan. 2024, www.kaggle.com/code/karltonkxb/food-recipe-association-rules. Accessed 07 December 2025.