Find the ”Chef”

Pedro Godinho - ist193608, Tiago Camarinhas - ist196769, Vasco Rodrigues - ist1100493, and Pedro Abreu - ist1112243

Group 51
Instituto Superior Técnico, Universidade de Lisboa

October 15, 2025

1 Introduction

This project tackles the classification of recipes to one of six chefs based on textual and metadata clues. The primary challenge was the dataset's imbalance and the effective combination of language understanding from a transformer model with hand-crafted structural features.

Our two-stage approach first uses a fine-tuned BERT model to generate chef predictions based on recipe text. These predictions are then combined with engineered features, such as ingredient counts, text lengths, and publication dates, and fed into a final classification network. This method leverages both semantic understanding and structural metadata to identify each chef's style.

2 Models

We initially tested a single-stage model that concatenated BERT embeddings with our engineered features, which achieved 85% accuracy. However, this approach led to very large input vectors, and made fine-tuning BERT impractical, leading us to instead adopt a two-stage pipeline.

Our final pipeline begins with **preprocessing**, where we extract features such as ingredient/tag count, text lengths, and publication year/day from the text fields. These features show different distributions per chef (Appendix A, Figures 1 to 5). For temporal features, the day of the year is encoded using sine and cosine transformations, to better represent the circularity of the feature. For numerical features, we apply min-max scaling.

In the first stage, a *bert-base-uncased* model[1] is fine-tuned on the combined recipe text. Its softmax outputs (chef probabilities) then serve as textual features. In the second stage, these BERT predictions are concatenated with the normalized features and used to train a Multi-Layer Perceptron (MLP) for the final classification. We made use of SMOTE to manage class imbalance during training.

3 Experimental Setup

3.1 Datasets

The dataset contains recipes from six chefs, which we split into stratified training (64%), validation (16%), and test (20%) sets. To address the class imbalance (Figure 6), we applied SMOTE to the training data.

[1] From the Hugging Face model hub: `https://huggingface.co/bert-base-uncased`

3.2 Metrics & Hyperparameters

Due to class imbalance, we used the F1 score as our primary guiding metric, supplemented by precision, recall, and confusion matrices, which were used to identify common misclassifications between chefs.

The final MLP uses one hidden layer (128 neurons), 0.001 as the L2 regularization parameter, 0.001 as the initial learning rate, ReLU activation, and Adam optimizer. These parameters were learned through a grid search over a variety of parameters.

4 Results

The fine-tuned BERT model alone achieved 86% validation accuracy, beating the first approach with off-the-shelf BERT embeddings. Our final two-stage model improved upon this, reaching a weighted F1-score of 0.91 on the hold-out test set. This confirms that the engineered features provided value beyond the text-only model. More detailed per-class metrics and the confusion matrix can be seen in tables 1 and 2, respectively. For brevity in these tables, we label the chefs as follows: C1 (1533), C2 (3288), C3 (4470), C4 (5060), C5 (6357), and C6 (8688).

Table 1: Classification Report for the final model.

Class	Precision	Recall	F1-Score	Support
C1	0.85	0.84	0.84	81
C2	0.92	0.90	0.91	90
C3	0.93	0.94	0.94	161
C4	0.95	0.93	0.94	107
C5	0.92	0.92	0.92	74
C6	0.87	0.91	0.89	87
Accuracy			0.91	600
Macro Avg	0.91	0.91	0.91	600
Weighted Avg	0.91	0.91	0.91	600

Table 2: Confusion Matrix for the final model.

	Predicted					
Actual	**C1**	**C2**	**C3**	**C4**	**C5**	**C6**
C1	68	1	5	1	2	4
C2	3	82	1	0	1	3
C3	5	1	152	2	0	1
C4	3	2	2	98	1	1
C5	1	0	2	0	69	2
C6	2	5	0	0	3	77

5 Discussion

The F1-score of 0.91 shows the model's effectiveness. Engineered features complement BERT by providing global context beyond its 512-token limit. For example, features like total text length or step count help the MLP resolve ambiguities when BERT assigns similar probabilities to multiple chefs.

The confusion matrix (Table 2) shows that the most frequent errors occur between Chefs C1 and C3. For example, "Canadian steak rub" by Chef C1 was misclassified as Chef C3. Both chefs share concise names (2-3 words), moderate ingredient counts (7 items), and minimal steps (3 steps). These chefs exhibit similar metadata patterns (as can be seen in the graphs in Appendix A). Notably, Chef C3 is the most represented class, but the model's high precision for this chef suggests our use of SMOTE mitigated prediction bias.

6 Future Work

Future work could focus on three areas. First, k-fold cross-validation for better hyperparameter tuning, though retraining BERT per fold is computationally expensive. Second, feature selection on BERT input maximizes its token limit. Finally, experimenting with different transformer models and additional engineered features, such as number of adjectives, punctuation, or sentence length.

Appendix A: Extra Figures and Tables

Figure 1: Box plot of recipe description length distribution for each chef.

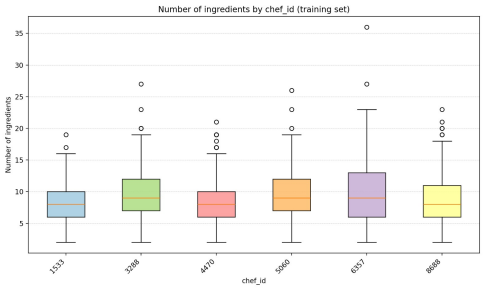

Figure 2: Box plot of the number of ingredients per recipe for each chef.

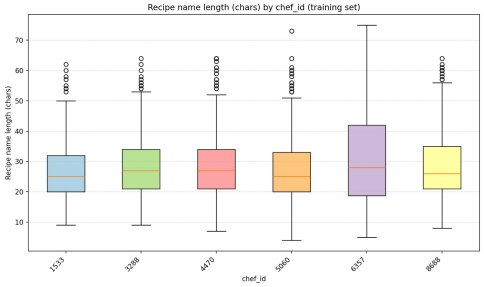

Figure 3: Box plot of recipe name length distribution for each chef.

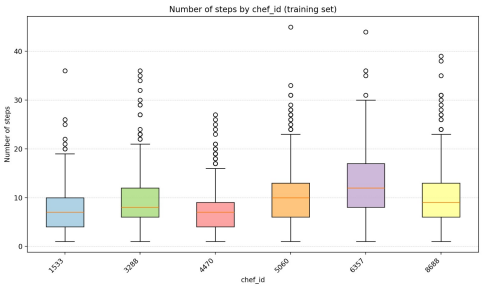

Figure 4: Box plot of the number of steps per recipe for each chef.

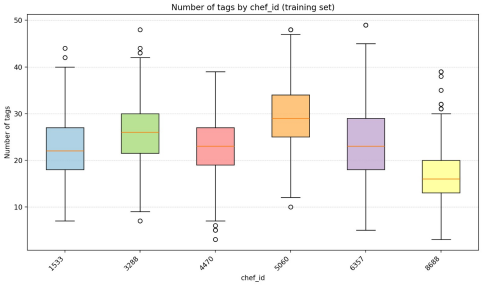

Figure 5: Box plot of the number of tags per recipe for each chef.

Figure 6: Bar chart of the number of recipes for each chef.