Identifying Key Entities in Recipe Data

1. Problem Statement

This project focuses on developing a **Named Entity Recognition (NER)** model using **Conditional Random Fields (CRF)** to extract meaningful entities from culinary recipe data. The primary goal is to automatically identify and classify text tokens into key categories such as **ingredients**, **quantities**, and **units**.

By converting unstructured recipe text into a structured format, this model can enable advanced applications in:

- Recipe management systems
- Dietary tracking tools
- E-commerce platforms (e.g., smart grocery lists)

The dataset comprises various culinary recipes, each with structured ingredient lists. These lists contain tokens labeled with their respective roles (e.g., "2" as quantity, "cups" as unit, "flour" as ingredient). This diversity supports the development of systems capable of understanding and analyzing culinary content effectively.

2. Methodology

The development of the NER system followed a multi-step methodology:

a. Data Preparation

- The dataset includes a collection of ingredient strings with labeled entities in **JSON** format.
- Each token in a string is annotated as one of the following:
    - Quantity
    - Unit
    - Ingredient

b. Text Preprocessing

- Techniques such as **regular expressions** and **token normalization** were used to clean and standardize the text.

- Additional preprocessing included **part-of-speech (POS)** tagging and **token splitting** for better structure.

c. Feature Engineering

Custom features were extracted for each token to improve model accuracy. These include:

- Lowercase transformation of the word
- Common word suffixes (e.g., "-ed", "-ly")
- Identification of numeric and fractional tokens
- Position-based context (previous and next words)
- Capitalization status and punctuation flags

d. Model Training

- A **CRF model** was implemented using the sklearn_crfsuite library.
- CRFs are ideal for sequence labeling because they account for **contextual dependencies** between adjacent labels.

e. Model Evaluation

The performance of the trained model was evaluated using:

- **Precision**
- **Recall**
- **F1-score**
- **Confusion matrix** and **classification report** for deeper error analysis

3. Techniques Used

The project leveraged techniques across multiple disciplines:

a. Natural Language Processing (NLP)

- Tokenization and POS tagging
- Custom entity labeling using domain-specific categories (e.g., quantity, unit, ingredient)

b. Machine Learning

- CRF for structured prediction in sequential data
- Flat classification metrics for evaluation

c. Data Engineering

- Conversion from JSON to tabular format (e.g., DataFrame)
- Data visualization using tools such as **Matplotlib** and **Seaborn**

d. Model Serialization

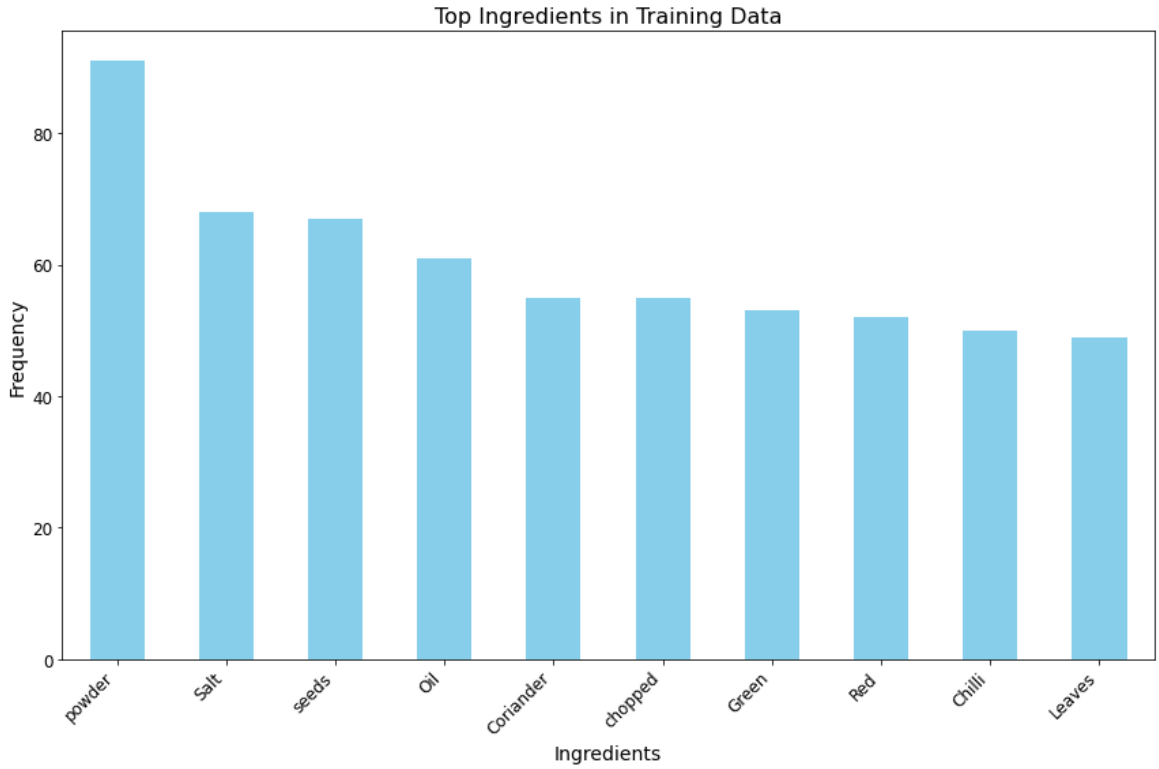
- The final model was saved using **joblib**, enabling easy deployment and reuse

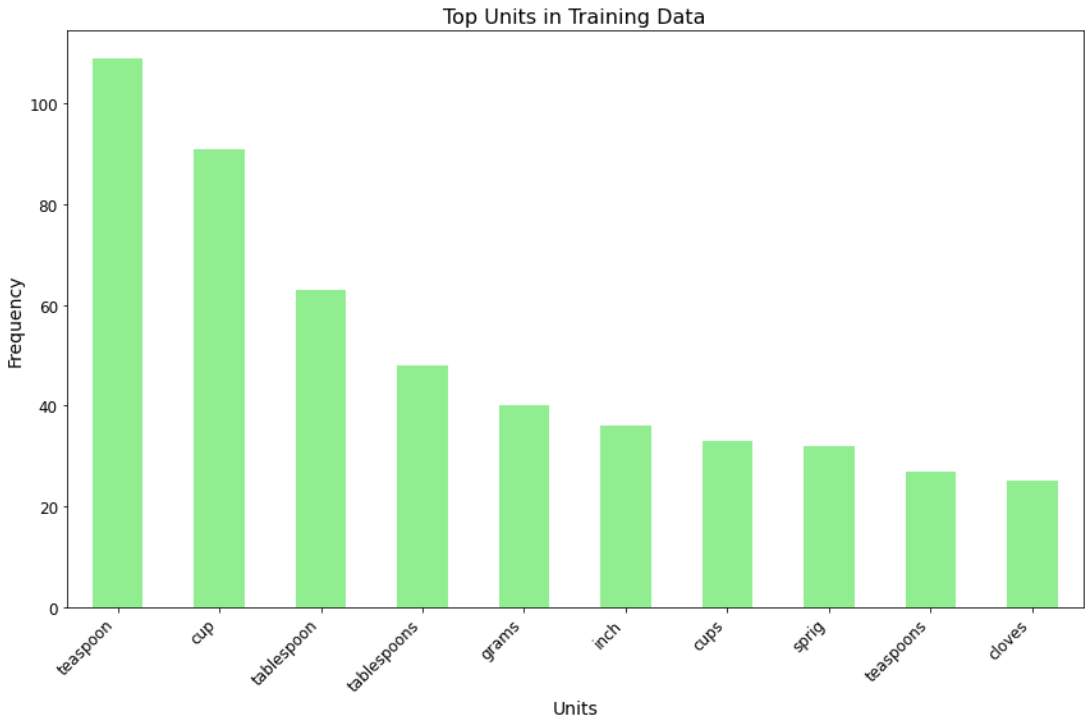
4. Key Insights

4.1 Data Visualization and Exploration

Token-level visualizations helped uncover:

- Frequency distribution of tokens
- Common misclassification patterns
- Label imbalance issues


Top Ingredients in Training Data
Frequency
0
20
40
60
80
powder
Salt
seeds
Oil
Coriander
chopped
Green
Red
Chilli
Leaves
Ingredients


Top Units in Training Data
Frequency
0
20
40
60
80
100
teaspoon
cup
tablespoon
tablespoons
grams
inch
cups
sprig
teaspoons
cloves
Units

4.2 Model Performance

- The trained CRF model achieved a high overall accuracy of **99.8%**, indicating strong performance in entity identification.

4.3 Evaluation Summary

Label Confusion:

- **Quantity vs. Unit:** Terms like "little" and "taste" were often incorrectly classified due to ambiguity and proximity to measurement terms.
- **Ingredient Misclassification:** Words like "taste" or auxiliary verbs (e.g., "is") were occasionally misclassified as ingredients.

Class Weight Adjustments:

- Balanced class weights were used to mitigate the impact of label imbalance.
- However, the ingredient label continued to be challenging, suggesting further tuning is needed.

Contextual Challenges:

- Some errors resulted from weak contextual signals.
- Words such as "per", "of", and "little" were not well-represented in the feature set, leading to confusion.

Sequence Boundary Tokens:

- The use of BOS (Beginning of Sentence) and EOS (End of Sentence) markers improved segmentation.
- Despite this, edge tokens in short or ambiguous sequences still presented classification issues.

Improvement Recommendations:

- Enhance feature representations by incorporating richer context
- Experiment with deep learning-based models such as **BiLSTM-CRF**
- Use domain-specific dictionaries for better coverage
- Conduct further analysis on adjectives and verbs commonly misclassified as ingredients

4.4 Conclusion

This project demonstrates that:

- **CRF-based NER models are effective** in extracting structured information from informal and domain-specific text such as recipes.
- Carefully designed **custom features**—including contextual and morphological patterns—significantly improve model accuracy.
- The methodology is **scalable** and can be adapted to other domains, such as:
    - Medical prescriptions
    - Product listings
    - Shopping lists

Structuring recipe data not only enhances information retrieval but also unlocks the potential for advanced use cases such as:

- Personalized recipe recommendations
- Automated grocery list generation
- Nutritional analysis and diet planning

The results validate the potential of combining NLP and CRF for building intelligent, domain-specific NER systems.