CRF-Based Ingredient Extraction Project Report

1. Introduction

This project focuses on the identification and classification of entities such as **ingredients**, **quantities**, and **units** from unstructured recipe text using a **Conditional Random Field (CRF)** model. Named Entity Recognition (NER) in the culinary domain is crucial for building structured recipe datasets, which can further be used in food recommendation systems, nutritional calculators, and inventory planners.

---

2. Data Preparation

- Input Data: A dataset of recipe ingredient lines.
- Preprocessing: Each sentence was tokenized and processed using spaCy. Each token was tagged with its Part-of-Speech (POS), lemma, and dependency information.
- Labels: Annotated labels were provided per token as ingredient, quantity, or unit.

---

3. Feature Engineering

Two core functions were created:

3.1 word2features(sent, i)

This function extracts rich token-level features for each token at index i in a given sentence sent.

Features Extracted:

- Core Features:
    - token, lemma, pos_tag, tag, dep, shape

    - Boolean indicators: is_stop, is_digit, has_digit, has_alpha, hyphenated, slash_present, is_title, is_upper, is_punct

- Quantity and Unit Detection:
    - Uses unit_keywords, quantity_keywords, and regex quantity_pattern to compute:
        - is_quantity, is_unit, is_numeric, is_fraction, is_decimal
- Contextual Features:
    - Previous and next tokens with:
        - prev_token, prev_is_quantity, prev_is_digit, BOS
        - next_token, next_is_unit, next_is_ingredient, EOS

3.2 sent2features(sent)

Applies word2features to all tokens in a sentence.

---

4. Dataset Transformation

- Features were extracted and stored in:
    - X_train_features and X_val_features
- Labels were converted to:
    - y_train_labels and y_val_labels
- Data was further flattened for analysis:
    - `y_train_flat` for label distribution

---

5. Class Weight Computation

To combat label imbalance, **inverse frequency-based class weights** were calculated:

- weight_dict[label] = total_samples / count(label)

To avoid overfitting to the dominant class (ingredient), its weight was manually penalized.

---

6. Weighted Feature Extraction

A function extract_features_with_class_weights was defined:

- Applies features and appends class_weight to each token.
- Final structures:
    - X_train_weighted_features, X_val_weighted_features
    - train_sample_weights, val_sample_weights

---

7. Model Training

CRF Model Configuration:

Parameter	Value
`algorithm`	'lbfgs'
`c1`	0.5
`c2`	1.0
`max_iterations`	100
`all_possible_transitions`	True

- CRF model trained on:
    - `X_train_weighted_features`

- `y_train_labels`
- `train_sample_weights`

---

8. Model Evaluation

8.1 On Training Data

- Classification Report

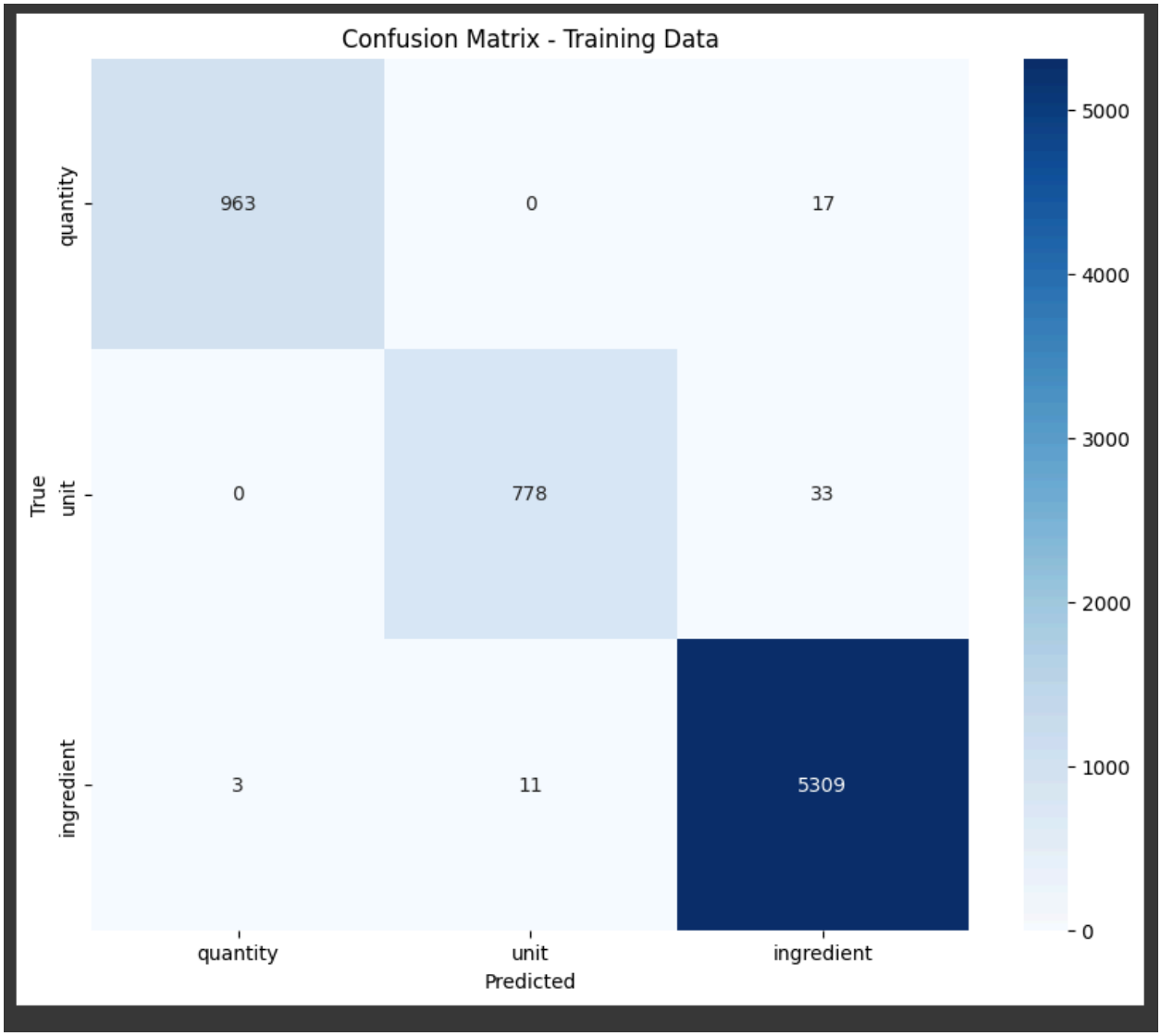
```
Training Set Evaluation

              precision    recall  f1-score   support


  ingredient       0.99      1.00      0.99      5323
    quantity       1.00      0.98      0.99       980
        unit       0.99      0.96      0.97       811


    accuracy                           0.99      7114
   macro avg       0.99      0.98      0.99      7114
weighted avg       0.99      0.99      0.99      7114
```

- Confusion Matrix



8.2 On Validation Data

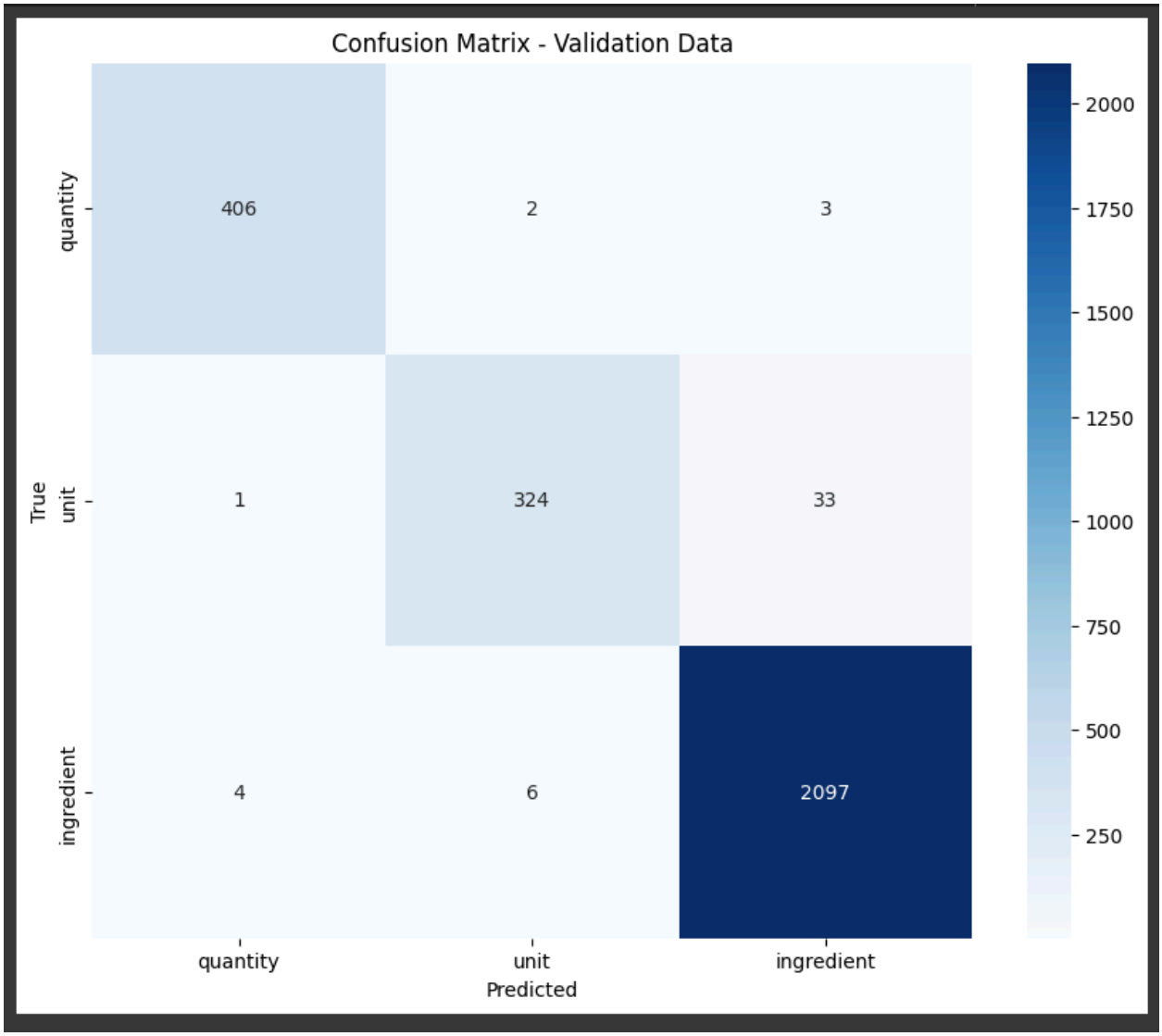
- Predictions: `y_pred_val`
- Classification Report & Confusion Matrix

```
Validation Set Evaluation - Classification Report

              precision    recall  f1-score   support


  ingredient       0.98      1.00      0.99      2107
```

quantity	0.99	0.99	0.99	411
unit	0.98	0.91	0.94	358
accuracy			0.98	2876
macro avg	0.98	0.96	0.97	2876
weighted avg	0.98	0.98	0.98	2876



- **Accuracy**: We have good accuracy on validation dataset as can be seen above

---

9. Error Analysis

9.1 Flattened Error Data

- Labels and predictions were flattened.
- Contextual information (`prev_token`, `next_token`, `class_weight`) was added.
- `error_data` DataFrame created.

9.2 Insights from Misclassifications

🧪 **Sample Errors:**

Token	True Label	Predicted	Notes
few	ingredient	quantity	Misinterpreted due to keyword overlap.
cloves	ingredient	unit	Confusion between similar vocabulary.
Spoon	unit	ingredient	Case-sensitive error.
gram	unit	ingredient	Missed quantity context.

---

10. Key Learnings & Recommendations

🔍 Issues Identified:

- Label-token misalignment may be present.
- High class imbalance.
- Feature representation may be insufficient or inconsistent.

🛠️ Recommendations:

- Re-check label alignment.

- Normalize casing and token formats.
- Augment training data with more samples for each class.
- Use additional context-aware models (e.g., BiLSTM-CRF or Transformer-based approaches).