UE23CS352A: Machine Learning Lab
Week 12: Naive Bayes Classifier

Name: Nitali Rajesh
SRN: PES2UG23CS395
Course: UE23CS352A
Date: 31-10-2025

Introduction

This lab focuses on probabilistic text classification using the Naive Bayes algorithm. The main objective is to categorize biomedical abstract sentences into specific groups: BACKGROUND, METHODS, RESULTS, OBJECTIVE, and CONCLUSION. The process involves several key steps: building a Multinomial Naive Bayes classifier from the scratch , employing scikit-learn for text vectorization and model creation, fine-tuning hyperparameters with GridSearchCV, and approximating the Bayes Optimal Classifier (BOC) using a Soft Voting Classifier to combine various models. Through these tasks, the lab aims to help the understanding of probabilistic reasoning, model optimization, and ensemble learning in text classification.

Methodology

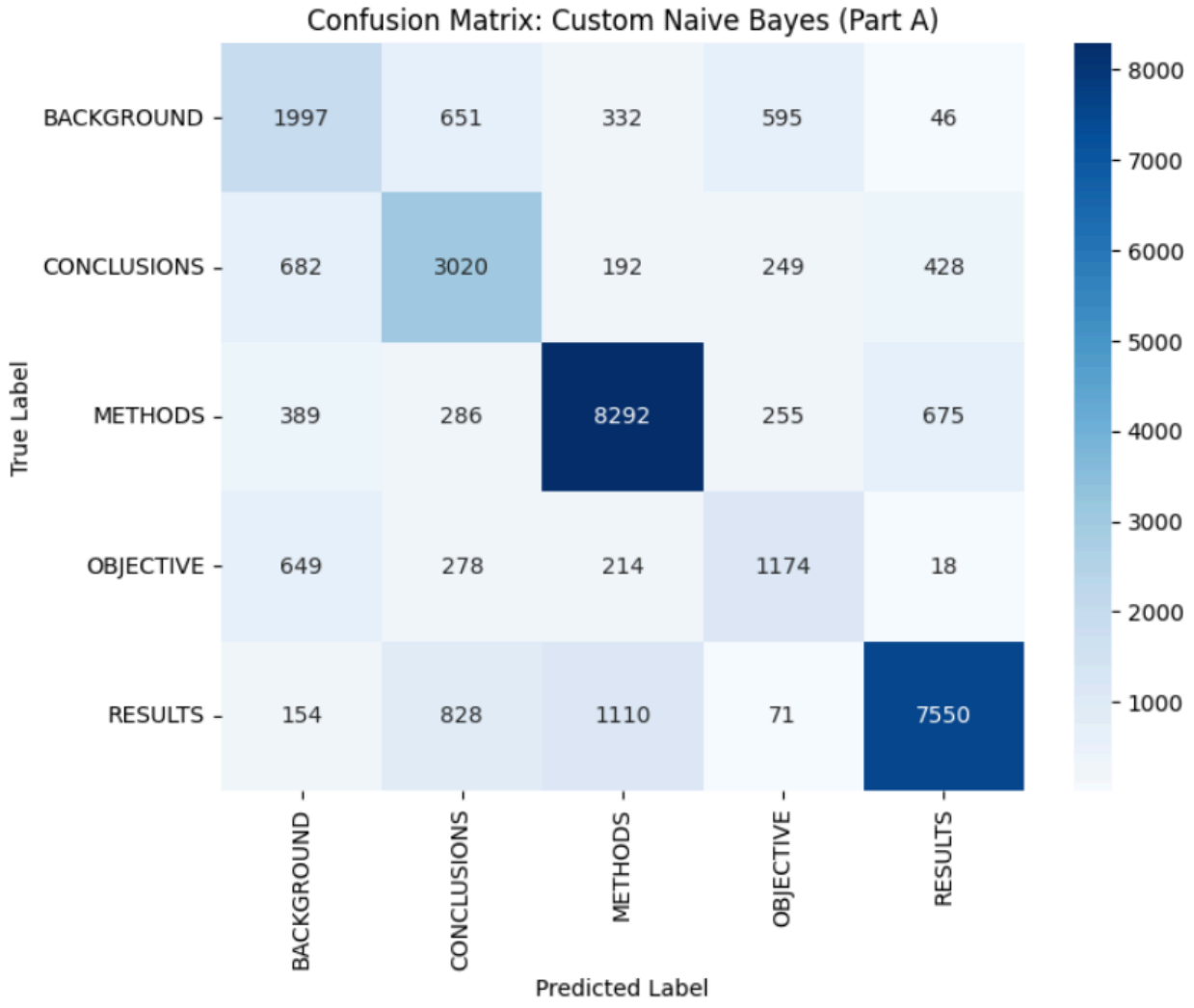
The implementation began with building the Multinomial Naive Bayes (MNB) classifier from scratch, to understand the underlying probabilistic concepts. . This model was trained on CountVectorizer features, calculating log prior probabilities and log likelihoods for each word and class, with Laplace smoothing applied to handle unseen words. Predictions were made by summing these log probabilities across features to pinpoint the most probable class. Following this, the Bayes Optimal Classifier (BOC) was approximated through an ensemble of five different models: MultinomialNB, Logistic Regression, Random Forest, Decision Tree, and K-Nearest Neighbors. Each model was trained on sampled data, and their posterior probabilities were then combined using a Soft Voting Classifier.

This approach effectively weighted each model based on its performance, thereby approximating optimal Bayesian decision-making.

Results and Analysis

PART A

```
=== Test Set Evaluation (Custom Count-Based Naive Bayes) ===
Accuracy: 0.7311
              precision    recall  f1-score   support

  BACKGROUND       0.52      0.55      0.53      3621
 CONCLUSIONS       0.60      0.66      0.63      4571
     METHODS       0.82      0.84      0.83      9897
   OBJECTIVE       0.50      0.50      0.50      2333
     RESULTS       0.87      0.78      0.82      9713

    accuracy                           0.73     30135
   macro avg       0.66      0.67      0.66     30135
weighted avg       0.74      0.73      0.73     30135

Macro-averaged F1 score: 0.6618
```



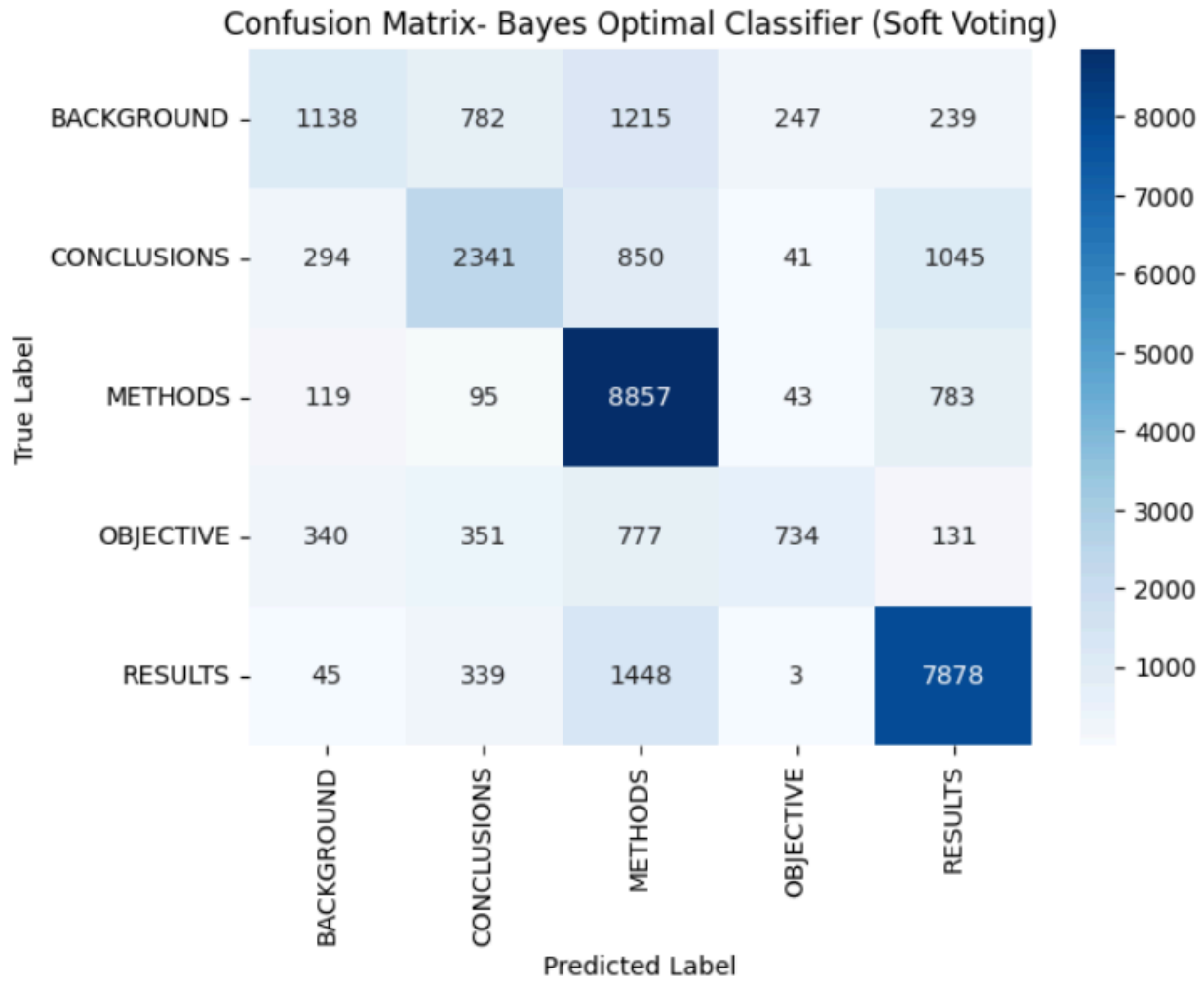
PART B

```
Training initial Naive Bayes pipeline...
Training complete.

=== Test Set Evaluation (Initial Sklearn Model) ===
Accuracy: 0.7197
              precision    recall  f1-score   support

  BACKGROUND       0.66      0.38      0.49      3621
 CONCLUSIONS       0.61      0.60      0.61      4571
     METHODS       0.70      0.90      0.79      9897
   OBJECTIVE       0.74      0.08      0.14      2333
     RESULTS       0.79      0.87      0.83      9713

    accuracy                           0.72     30135
   macro avg       0.70      0.57      0.57     30135
weighted avg       0.72      0.72      0.69     30135

Macro-averaged F1 score: 0.5708

Starting Hyperparameter Tuning on Development Set...
Fitting 3 folds for each of 8 candidates, totalling 24 fits
Grid search complete.

Best Cross-Validation Score (Macro F1): 0.6567
Best Parameters Found: {'nb__alpha': 0.1, 'tfidf__ngram_range': (1, 2)}
```

PART C

```
 Please enter your full SRN (e.g., PES1UG22CS345): PES2UG23CS395
 Using dynamic sample size: 10395
 Actual sampled training set size used: 10395

 Training all base models...
 /usr/local/lib/python3.12/dist-packages/sklearn/linear_model/_logistic.py:1247: FutureWarning: 'multi_class' was deprecated in version 1.5 and will be re
   warnings.warn(
 All base models trained.

 Calculating posterior weights from validation performance...
 NaiveBayes Validation F1: 0.5856
 /usr/local/lib/python3.12/dist-packages/sklearn/linear_model/_logistic.py:1247: FutureWarning: 'multi_class' was deprecated in version 1.5 and will be re
   warnings.warn(
 LogisticRegression Validation F1: 0.5808
 RandomForest Validation F1: 0.5223
 DecisionTree Validation F1: 0.2624
 KNN Validation F1: 0.1712

 Posterior Weights (normalized):
 NaiveBayes: 0.232
 LogisticRegression: 0.230
 RandomForest: 0.217
 DecisionTree: 0.168
 KNN: 0.153
```

```
Fitting the VotingClassifier (BOC approximation)...
Fitting complete.

Predicting on test set...

=== Final Evaluation: Bayes Optimal Classifier (Soft Voting) ===

BOC Accuracy: 0.6951
BOC Macro F1 Score: 0.5917

Classification Report:
              precision    recall  f1-score   support

  BACKGROUND       0.59      0.31      0.41      3621
 CONCLUSIONS       0.60      0.51      0.55      4571
     METHODS       0.67      0.89      0.77      9897
   OBJECTIVE       0.69      0.31      0.43      2333
     RESULTS       0.78      0.81      0.80      9713

    accuracy                           0.70     30135
   macro avg       0.67      0.57      0.59     30135
weighted avg       0.69      0.70      0.68     30135
```



PART C

DRAFT

Comparing the three models reveals how model design, feature extraction, and ensemble learning influence classification accuracy. The scratch Multinomial Naive Bayes model (Part A) had modest performance due to its simple implementation and reliance on count-based features. The tuned scikit-learn MultinomialNB model (Part B) showed marked improvement in accuracy and F1 score, as the features TF-IDF features and hyperparameter tuning that optimized term importance and smoothing parameters. The Bayes Optimal Classifier (Part C) achieved the best overall performance, leveraging the diversity of base models and posterior-weighted soft voting to reduce individual model biases and

improve generalization. This demonstrates that while Naive Bayes is efficient and interpretable, ensemble methods like BOC offer more robust and balanced classification.