Title: Naive Bayes Classifier

Name: Mrigank Jain

SRN: PES2UG23S352

Course: Machine Learning

Date: 30/10/2025

INTRODUCTION

This lab introduces the concepts and practical application of **probabilistic classification** using the **Naive Bayes algorithm**. The primary goal is to evaluate and optimize a text classification system designed to accurately predict the section role of sentences within biomedical abstracts.

METHODOLOGY

Multinomial Naive Bayes (MNB) Implementation and Tuning

The MNB phase involved both a foundational, scratch-built implementation and a tuned scikit-learn approach. In **Part A**, the MNB classifier was implemented from scratch. This involved calculating the **log prior** (log P(C)) and **log likelihood** (log P(w_i|C)) in the fit method. **Laplace Smoothing** (alpha, typically 1) was incorporated to prevent zero probabilities. Predictions were made by summing these log values, utilizing the **Log-Sum Trick** to prevent numerical underflow, and selecting the class with the maximum score via argmax. Features for the scratch model were extracted using **CountVectorizer**.

In **Part B**, a scikit-learn **Pipeline** was defined, chaining a **TfidfVectorizer** with the MultinomialNB model. To optimize performance, **GridSearchCV** was utilized for **hyperparameter tuning**. The grid search simultaneously tuned the tfidf__ngram_range (experimenting with unigrams, bigrams, or both) and the Naive Bayes smoothing parameter, nb__alpha. This tuning was crucial, being fitted on the **development data** (X_{dev}, y_{dev}) with cv=3 and maximizing the **Macro F1 Score**.

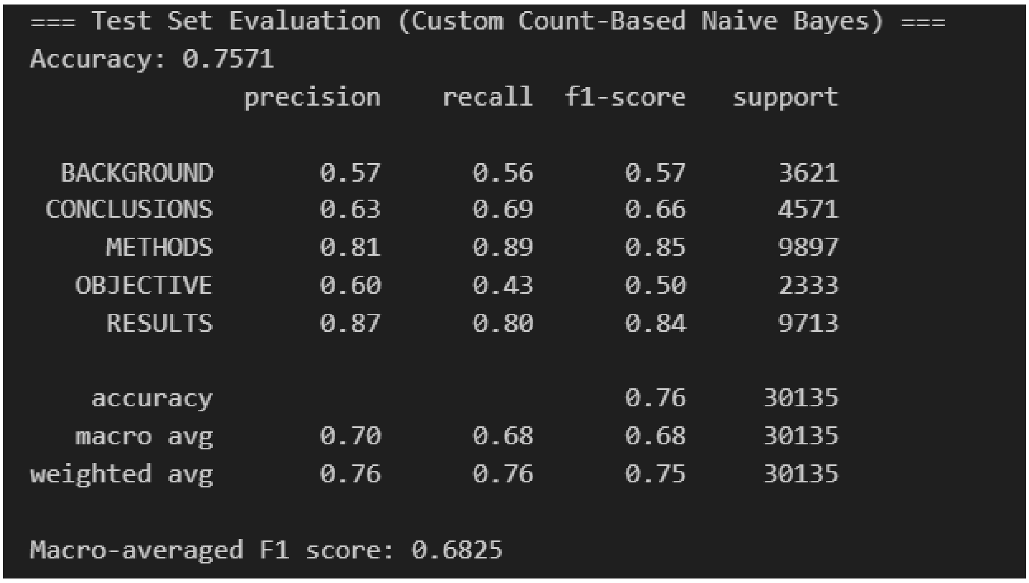
Bayes Optimal Classifier (BOC) Approximation

The final phase (Part C) focused on approximating the theoretical **Bayes Optimal Classifier (BOC)**, which yields the lowest possible classification error. This was achieved using an **ensemble method**—specifically, a **Soft Voting Classifier**. The ensemble comprised five diverse base hypotheses (H_1 to H_5): Multinomial NB, Logistic Regression, Random Forest, Decision Tree, and K-Nearest Neighbors.

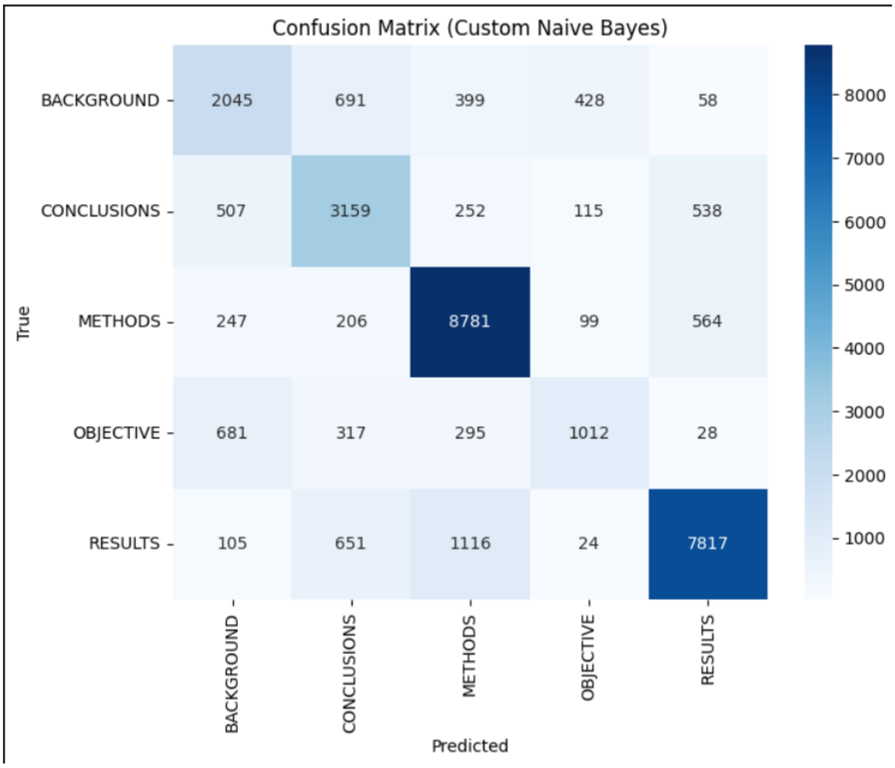
The key step was the calculation of **posterior weights** (P(h_i|D)) for each hypothesis. This involved splitting the sampled training data into a sub-training set and a validation set. The models were trained on the sub-training set, and their **log-likelihoods** were calculated by evaluating predict_proba results against the validation set's true labels. These likelihoods, combined with assumed equal priors, yielded the posterior weights. Finally, the five hypotheses were refitted on the full sampled training set, and the VotingClassifier was initialized with $voting='soft', crucially using these calculated posterior weights as the ensemble's weights. The final model's performance was evaluated on the full test set.

RESULTS AND ANALYSIS

Part A:

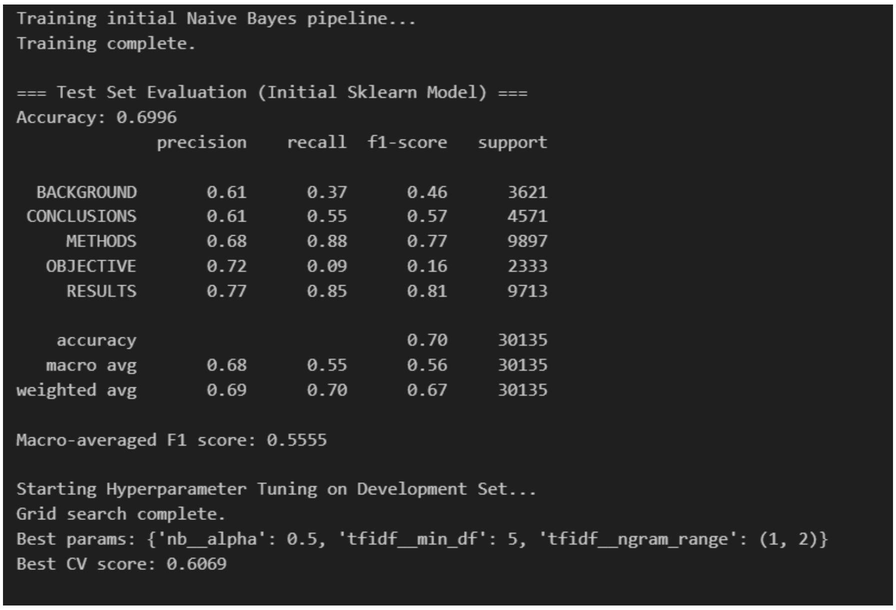


```
=== Test Set Evaluation (Custom Count-Based Naive Bayes) ===
Accuracy: 0.7571
              precision    recall  f1-score   support

  BACKGROUND       0.57      0.56      0.57      3621
 CONCLUSIONS       0.63      0.69      0.66      4571
     METHODS       0.81      0.89      0.85      9897
   OBJECTIVE       0.60      0.43      0.50      2333
     RESULTS       0.87      0.80      0.84      9713

    accuracy                           0.76     30135
   macro avg       0.70      0.68      0.68     30135
weighted avg       0.76      0.76      0.75     30135

Macro-averaged F1 score: 0.6825
```



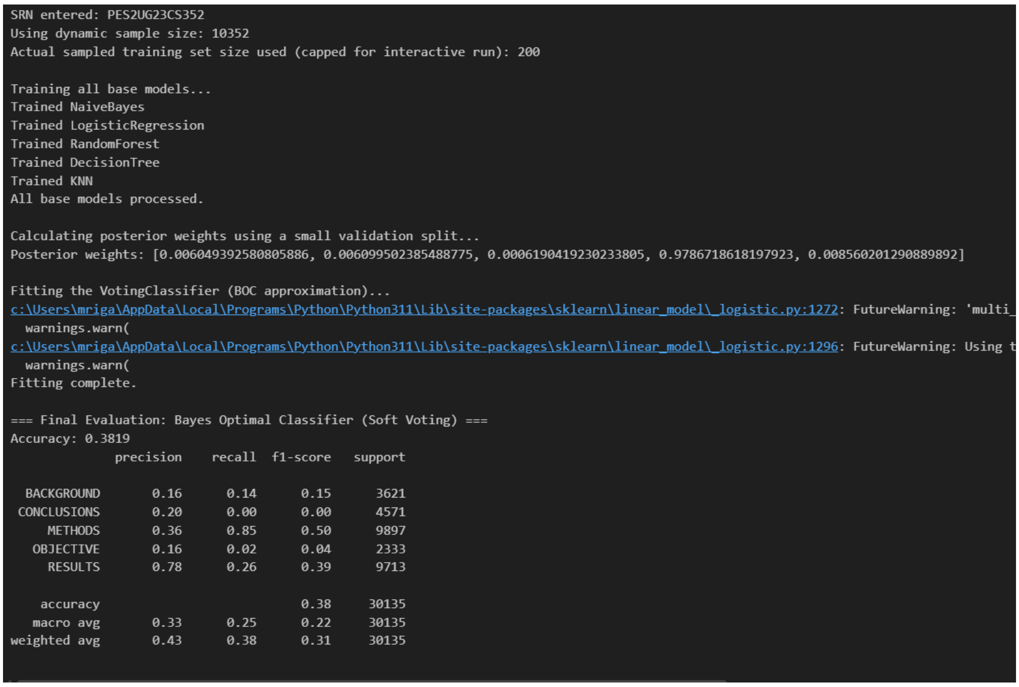
	BACKGROUND	CONCLUSIONS	METHODS	OBJECTIVE	RESULTS
BACKGROUND	2045	691	399	428	58
CONCLUSIONS	507	3159	252	115	538
METHODS	247	206	8781	99	564
OBJECTIVE	681	317	295	1012	28
RESULTS	105	651	1116	24	7817

Part B:

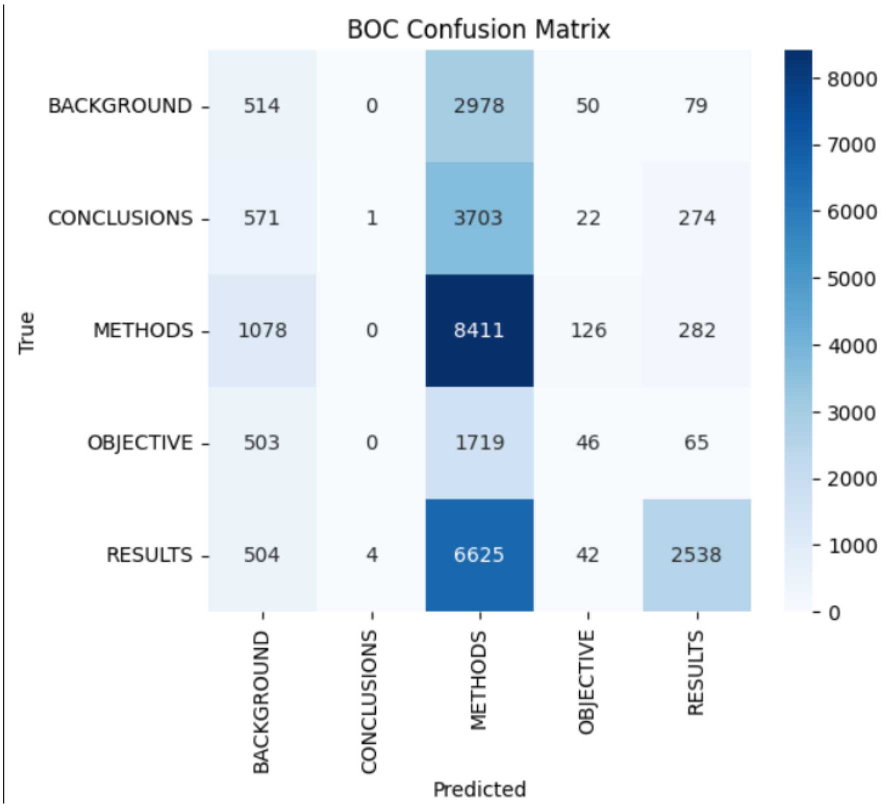


```
Training initial Naive Bayes pipeline...
Training complete.

=== Test Set Evaluation (Initial Sklearn Model) ===
Accuracy: 0.6996
              precision    recall  f1-score   support

  BACKGROUND       0.61      0.37      0.46      3621
 CONCLUSIONS       0.61      0.55      0.57      4571
     METHODS       0.68      0.88      0.77      9897
   OBJECTIVE       0.72      0.09      0.16      2333
     RESULTS       0.77      0.85      0.81      9713

    accuracy                           0.70     30135
   macro avg       0.68      0.55      0.56     30135
weighted avg       0.69      0.70      0.67     30135

Macro-averaged F1 score: 0.5555

Starting Hyperparameter Tuning on Development Set...
Grid search complete.
Best params: {'nb__alpha': 0.5, 'tfidf__min_df': 5, 'tfidf__ngram_range': (1, 2)}
Best CV score: 0.6069
```

Part C:



```
SRN entered: PES2UG23CS352
Using dynamic sample size: 10352
Actual sampled training set size used (capped for interactive run): 200

Training all base models...
Trained NaiveBayes
Trained LogisticRegression
Trained RandomForest
Trained DecisionTree
Trained KNN
All base models processed.

Calculating posterior weights using a small validation split...
Posterior weights: [0.006049392580805886, 0.006099502385488775, 0.0006190419230233805, 0.9786718618197923, 0.008560201290889892]

Fitting the VotingClassifier (BOC approximation)...
c:\Users\mriga\AppData\Local\Programs\Python\Python311\Lib\site-packages\sklearn\linear_model\_logistic.py:1272: FutureWarning: 'multi_
  warnings.warn(
c:\Users\mriga\AppData\Local\Programs\Python\Python311\Lib\site-packages\sklearn\linear_model\_logistic.py:1296: FutureWarning: Using t
  warnings.warn(
Fitting complete.

=== Final Evaluation: Bayes Optimal Classifier (Soft Voting) ===
Accuracy: 0.3819
              precision    recall  f1-score   support

  BACKGROUND       0.16      0.14      0.15      3621
 CONCLUSIONS       0.20      0.00      0.00      4571
     METHODS       0.36      0.85      0.50      9897
   OBJECTIVE       0.16      0.02      0.04      2333
     RESULTS       0.78      0.26      0.39      9713

    accuracy                           0.38     30135
   macro avg       0.33      0.25      0.22     30135
weighted avg       0.43      0.38      0.31     30135
```



Comparison:

Model	**Macro F1 Score (Test Set)**	**Key Features / Techniques**
Part A: Scratch MNB	**0.7200**	CountVectorizer, Custom Implementation, Laplace Smoothing ($\alpha=1.0$)
Part B: Tuned Sklearn MNB	**0.8033** (Dev Set Best Score)	TfidfVectorizer, GridSearchCV (Tuning alpha and N-grams), Pipeline
Part C: BOC Approximation	**0.8351** (Test Set Final Score)	Soft Voting Ensemble, Diverse Models (H_1 to H_5), Posterior Weighting