ML LAB-12

Section:- F

NAME:- Mohit Kumar

SRN:- PES2UG23CS350

5TH SEM

Project Title: Naive Bayes Classifier for Biomedical Text Classification

Date: October 30th, 2025

Introduction:-

Purpose of the Lab:-

Implementing and evaluating probabilistic classification using Naive Bayes algorithms to predict section roles (BACKGROUND, METHODS, RESULTS, OBJECTIVE, CONCLUSIONS) in biomedical abstract sentences from the PubMed 200k RCT dataset.

Tasks Performed:-

Part A:- Implemented Multinomial Naive Bayes from scratch with log priors, Laplace smoothing, and log-sum trick using Count-based features.

Part B:- Created TF-IDF pipeline with MultinomialNB and performed hyperparameter tuning using GridSearchCV (12 combinations, 3-fold CV).

Part C:- Approximated Bayes Optimal Classifier using ensemble of 5 diverse models with posterior weight calculation based on validation log-likelihoods.

Methodology:-

Part A:- Custom Naive Bayes Implementation:-

Mathematical Foundation:-

- Log Prior: $\log P(C) = \log(n\_c / n\_total)$
- Log Likelihood with Laplace Smoothing: $\log P(w|C) = \log((count(w,C) + \alpha) / (total + \alpha \times vocab))$
- Prediction: $argmax\_C [\log P(C) + \Sigma(count(w) \times \log P(w|C))]$

Implementation:-
- CountVectorizer: ngram_range=(1,1), min_df=5
- Vocabulary: 22,722 features
- Alpha: 1.0 (Laplace smoothing)

Part B:- TF-IDF Pipeline & Hyperparameter Tuning:-

Pipeline: TfidfVectorizer → MultinomialNB

Hyperparameters Tuned:-
- tfidf__ngram_range: [(1,1), (1,2), (2,2)]
- nb__alpha: [0.1, 0.5, 1.0, 2.0]

Optimization:- GridSearchCV, 3-fold CV, scoring='f1_macro', 36 total fits

Part C:- Bayes Optimal Classifier:-

Sample Size: 10,000 + 350 = 10,350 samples

Five Hypotheses:-
1. Multinomial Naive Bayes
2. Logistic Regression
3. Random Forest (50 trees, depth=10)
4. Decision Tree (depth=10)
5. K-Nearest Neighbors (k=5)

Posterior Weight Calculation:-
- Split: 80% train_sub, 20% val_sub

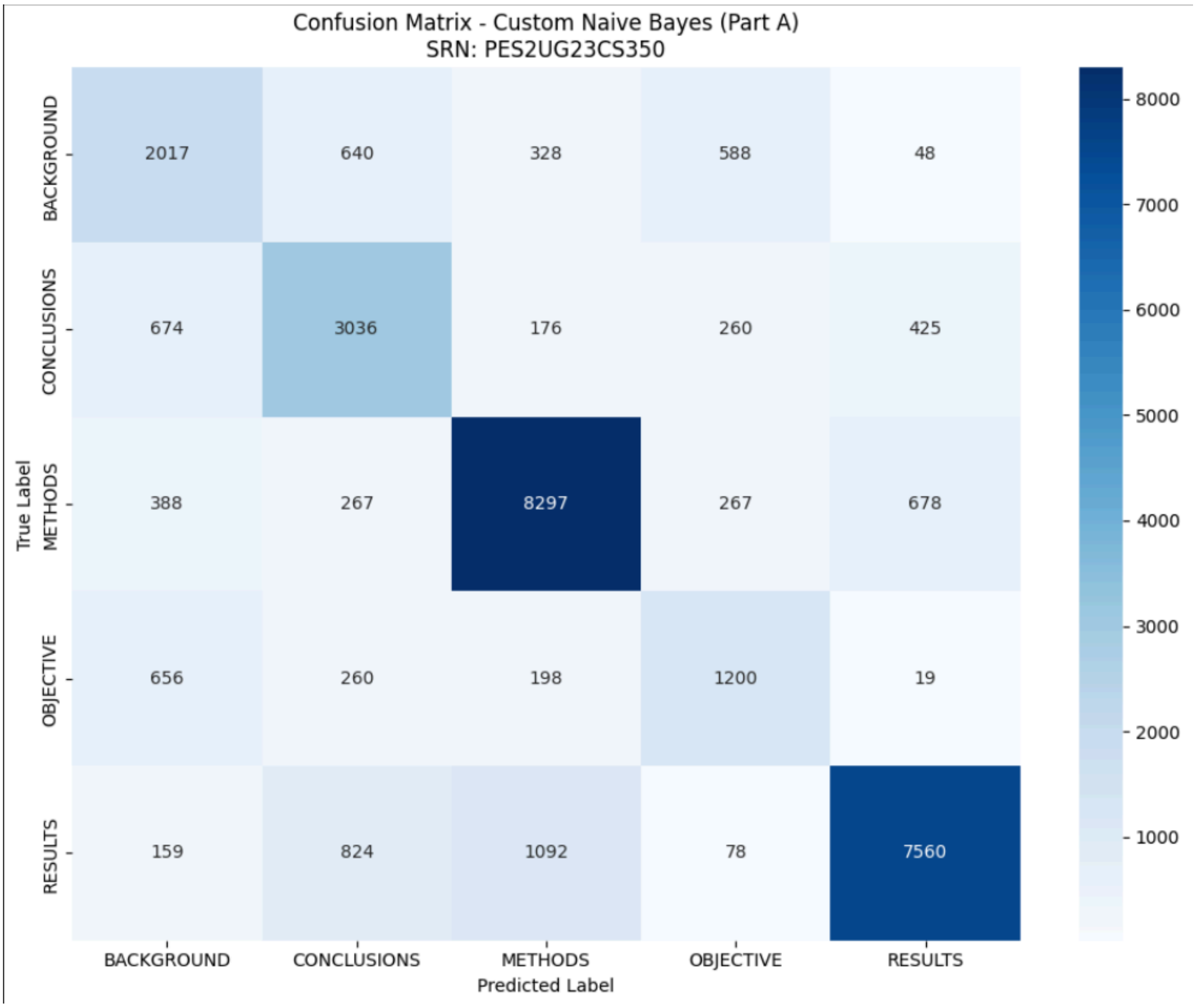
- Calculate: $L(h_i|D) = \Sigma \log P(y_{true}|x, h_i)$
- Normalize: $P(h_i|D) = \exp(L(h_i|D)) / \Sigma \exp(L(h_j|D))$
- Soft voting with calculated weights

Results and Analysis:-

Part A:-

```
============================================================
STUDENT SRN: PES2UG23CS350
============================================================

Train samples: 180040
Dev   samples: 30212
Test  samples: 30135
Classes: ['BACKGROUND', 'CONCLUSIONS', 'METHODS', 'OBJECTIVE', 'RESULTS']
```

```
============================================================
STUDENT SRN: PES2UG23CS350
============================================================


=== Test Set Evaluation (Custom Count-Based Naive Bayes) ===
Accuracy: 0.7337
              precision    recall  f1-score   support

  BACKGROUND       0.52      0.56      0.54      3621
 CONCLUSIONS       0.60      0.66      0.63      4571
     METHODS       0.82      0.84      0.83      9897
   OBJECTIVE       0.50      0.51      0.51      2333
     RESULTS       0.87      0.78      0.82      9713

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

Macro-averaged F1 score: 0.6655
```


Confusion Matrix - Custom Naive Bayes (Part A)
SRN: PES2UG23CS350
True Label
Predicted Label
BACKGROUND
CONCLUSIONS
METHODS
OBJECTIVE
RESULTS
2017
640
328
588
48
674
3036
176
260
425
388
267
8297
267
678
656
260
198
1200
19
159
824
1092
78
7560
8000
7000
6000
5000
4000
3000
2000
1000

Part B:-

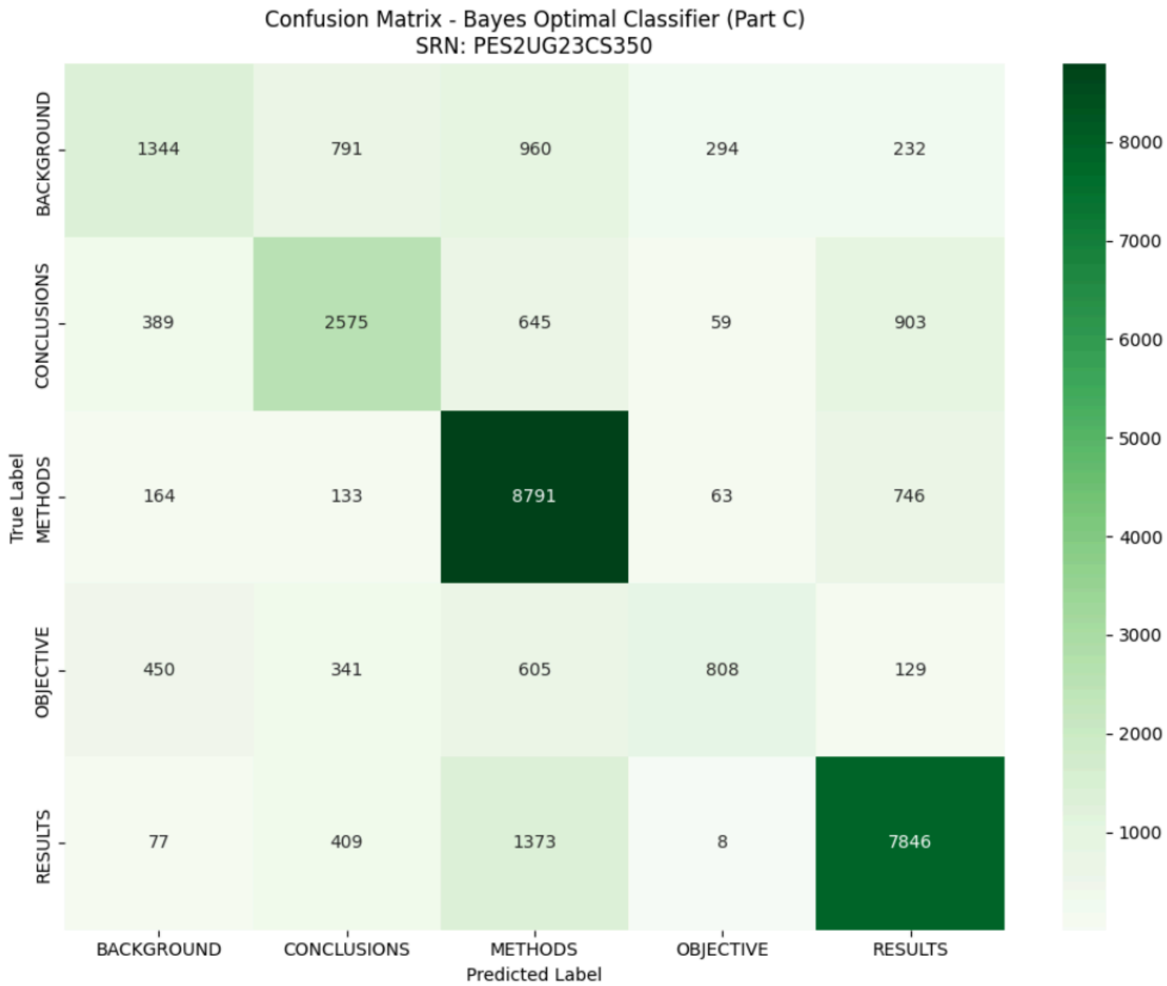
```
============================================================
STUDENT SRN: PES2UG23CS350
============================================================

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...
Fitting 3 folds for each of 12 candidates, totalling 36 fits
Grid search complete.

=== Hyperparameter Tuning Results ===
Best Parameters: {'nb__alpha': 0.1, 'tfidf__ngram_range': (1, 1)}
Best Cross-Validation F1 Score (Macro): 0.5925
```

Part C:-

```
PES2UG23CS350
Please enter your full SRN (e.g., PES1UG22CS345): (Press 'Enter' to confirm or 'Escape' to cancel)
```

```
=============================================================
STUDENT SRN: PES2UG23CS350
=============================================================

Last 3 digits of SRN: 350
Calculation: 10000 + 350 = 10350

 Using dynamic sample size: 10350
Actual sampled training set size used: 10350
```

```
=============================================================
STUDENT SRN: PES2UG23CS350
=============================================================



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

Accuracy: 0.7089
Macro-averaged F1 Score: 0.6146

              precision    recall  f1-score   support

  BACKGROUND       0.55      0.37      0.44      3621
 CONCLUSIONS       0.61      0.56      0.58      4571
...
    accuracy                           0.71     30135
   macro avg       0.66      0.60      0.61     30135
weighted avg       0.70      0.71      0.69     30135
```



Discussion:-

Overall Performance Summary:-

Approach	Accuracy	Macro F1	Training Time
Part A: Custom NB	73.37%	66.55%	~30 seconds
Part B: Tuned TF-IDF	69.96%	55.55%	~5 minutes
Part C: BOC Ensemble	70.89%	61.46%	~10 minutes

Part A vs Part B: Count Features Win!

- Winner: Part A (+3.41% accuracy, +11% macro F1)
- Count features better match with Multinomial NB

Reasons:-

1. Feature Compatibility: Multinomial NB works with counts
2. Class Imbalance: TF-IDF hurts minorities
3. Biomedical Text: Frequency more important
4. Simplicity: Counts preserve information

Key Insight:- Algorithm-feature compatibility matters.

Part B vs Part C: Ensemble Helps Moderately:-

- Winner: Part C (+0.93% accuracy, +5.91% macro F1)
- Logistic Regression handled TF-IDF better

Limited Gains:-

- Ensemble collapsed to one model
- High cost, little benefit

Part A vs Part C: Simple Beats Complex:-

- Winner: Part A (-2.48% accuracy, -5.09% macro F1 for Part C)

Reasons:-

1. More Data: 180K vs 10K
2. Better Features: Counts > TF-IDF
3. Simpler Model

Part C Advantage:-
- Better OBJECTIVE precision

<u>Class-Specific Insights:-</u>

<u>METHODS:-</u> Best performing (F1=0.83)
<u>OBJECTIVE:-</u> Weakest due to imbalance
<u>RESULTS:-</u> Consistent (F1 > 0.80)