Full results for best case this far:

vectorizer = TfidfVectorizer(min\_df=2, ngram\_range=(1,2),

                preprocessor=my\_preprocessor, stop\_words=imdb\_stop\_words)

Train Accuracy: 0.9903948772678762

Test Accuracy: 0.8434504792332268

Train f1 Score: 0.9903121636167922

Test f1 Score: 0.8327645051194539

ROC\_AUC Score: 0.8436632369753388

Classification Report:

prec recall f1-sc support

-1 0.80 0.91 0.85 156

1 0.90 0.78 0.83 157

accuracy 0.84 313

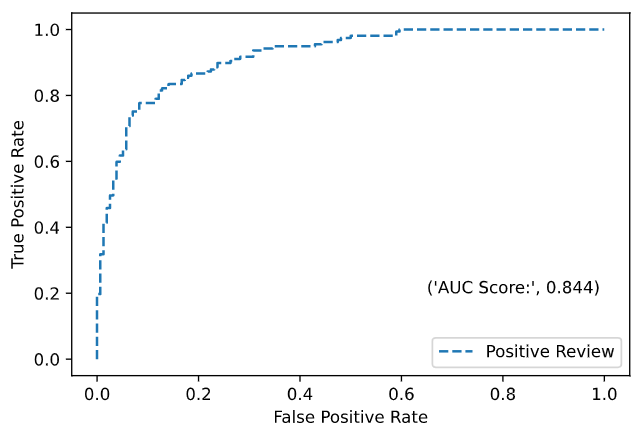
macro avg 0.85 0.84 0.84 313

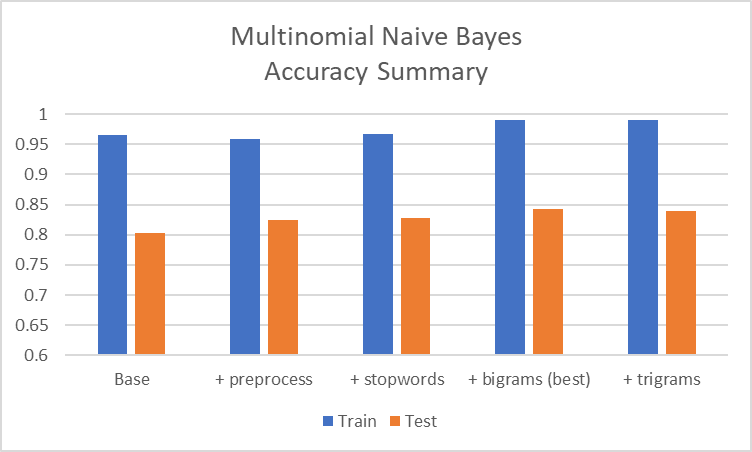
weighted avg 0.85 0.84 0.84 313

Confusion matrix:

[142 14]

[ 35 122]





Summaries:

Basic case (defaults)

vectorizer = TfidfVectorizer(min\_df=2)

Train Accuracy: 0.9658484525080042

Test Accuracy: 0.8019169329073482

Confusion matrix:

[[140 16]

[ 46 111]]

Add preprocessor

vectorizer = TfidfVectorizer(min\_df=2, preprocessor=my\_preprocessor)

#...

Train Accuracy: 0.9583778014941302

Test Accuracy: 0.8242811501597445

Confusion matrix:

[[141 15]

[ 40 117]]

Add stop words

vectorizer = TfidfVectorizer(min\_df=2, preprocessor=my\_preprocessor,

            stop\_words=imdb\_stop\_words)

Train Accuracy: 0.9669156883671292

Test Accuracy: 0.8274760383386581

Confusion matrix:

[[140 16]

[ 38 119]]

Add bigrams

vectorizer = TfidfVectorizer(min\_df=2, preprocessor=my\_preprocessor,

            stop\_words=imdb\_stop\_words, ngram\_range=(1,2))

Train Accuracy: 0.9903948772678762

Test Accuracy: 0.8434504792332268

Confusion matrix:

[[142 14]

[ 35 122]]

Keep bigrams but remove stop words

vectorizer = TfidfVectorizer(min\_df=2, preprocessor=my\_preprocessor,

            stop\_words=None, ngram\_range=(1,2))

Train Accuracy: 0.9861259338313767

Test Accuracy: 0.8115015974440895

Confusion matrix:

[[140 16]

[ 43 114]]

Add bigrams and trigrams (decreases performance!)

vectorizer = TfidfVectorizer(min\_df=2, preprocessor=my\_preprocessor,

            stop\_words=imdb\_stop\_words, ngram\_range=(1,3))

Train Accuracy: 0.9903948772678762

Test Accuracy: 0.8402555910543131

Confusion matrix:

[[138 18]

[ 32 125]]

ngrams 1-4 (exactly same as (1-2)) – same for 1-5

vectorizer = TfidfVectorizer(min\_df=2, preprocessor=my\_preprocessor,

            stop\_words=imdb\_stop\_words, ngram\_range=(1,4))

Train Accuracy: 0.9903948772678762

Test Accuracy: 0.8434504792332268

Confusion matrix:

[[139 17]

[ 32 125]]

Only bigrams

vectorizer = TfidfVectorizer(min\_df=2, preprocessor=my\_preprocessor,

            stop\_words=imdb\_stop\_words, ngram\_range=(2,2))

#...

Train Accuracy: 1.0

Test Accuracy: 0.7412140575079872

Confusion matrix:

[[125 31]

[ 50 107]]

Bigrams and Trigrams only

vectorizer = TfidfVectorizer(min\_df=2, preprocessor=my\_preprocessor,

            stop\_words=imdb\_stop\_words, ngram\_range=(2,3))

#...

Train Accuracy: 1.0

Test Accuracy: 0.7476038338658147

Confusion matrix:

[[125 31]

[ 48 109]]