Report for Assignment 2

1. Result table

|  |  |  |
| --- | --- | --- |
| Stopwords removed | Text features | Accuracy (test set) |
| yes | Unigrams | 0.8058 |
| yes | Bigrams | 0.7898875 |
| yes | Unigram+bigrams | 0.8235875 |
| no | Unigrams | 0.8158125 |
| no | Bigrams | 0.869275 |
| no | Unigrams+bigrams | 0.87465 |

1. Answers
   1. Which condition performed better: with or without stopwords? Write a brief

paragraph (5-6 sentences) discussing why you think there is a difference in

performance.

The version with the stop words performs better for all n-gram models. The possible reasons would be stop words in nltk includes negative words like ‘no’, ‘never’ and ‘not’. When the downstream is sensitive to the negative words like sentiment analysis in our case. It is wise not to remove the negative words so as to preserve as much information as possible.

* 1. Which condition performed better: unigrams, bigrams or unigrams+bigrams?

Briefly (in 5-6 sentences) discuss why you think there is a difference?

In the version of texts with stop words, the bigram outperforms the unigrams by around 6% and the unigram+bigrams outperforms the bigram model slightly by 0.5%. The reason is the bigram model contains much more information, for example, ‘not good’ and ‘not like’ mean a negative sentiment in bigram model while in unigram, ‘not’ and ‘good’ is separated and ‘good’ can appear in both negative and positive documents, which can confused the model.

In the version of texts without stop words, the unigram actually outperforms the bigrams since we had the stop words removed, there is no reason to combine the tokens. The only bigram model may have much more noises than the unigram one . unigram and bigram still outperform the previous one since it has richer information.