Multinomial Event Model Naive Bayes

Steps to Compute Posterior Probabilities for Text Classification

Given an

- an input document x = <w1,w2...wn>
- the category that this document belongs to c

Training Time

- increment the count of total documents we have learned from N.
- increment the count of documents that have been mapped to this category Nc.
- if we encounter new words in this document, add them to our vocabulary, and update our vocabulary size |V|.
- update count(w, c) => the frequency with which each word in the document has been mapped to this category.
- update count (c) => the total count of all words that have been mapped to this class.

Testing Time
Compute Prior Prob of each class c -

P(c) = Nc / N

- Nc : No of documents having class c
- N : Total no of documents during training time

Likelihood (P(X|Y=c))
We need to iterate through each word in the document and calculate:
P(w | c) = [count(w, c) + 1] / [count(c) + |V|]

This is the count of how many times this word has appeared in class c, plus 1, divided by the total count of all words that have ever been mapped to class c, plus the vocabulary size. This uses the Laplace-Smoothing, so we don't get tripped up by words we've never seen before. This equation is used both for words we have seen, as well as words we haven't seen.

x = <w1,w2,w3....wn>
P(x|c) = Product(P(wi|c))

- We multiply each P(w | c) for each word w in the new document, then multiply by P(c

), and the result is the probability that this document belongs to this class.

Final Prediction
Predict the class which has highest P(Y=C|x) posterior probabiliy.