Introduction

The first section of today's class (8/13/15) was on the Bag of Words model representation of text data. The following notes detail the theory discussed in class.

Bag of Words Representation

A document is represented as a vector of words or token counts, where the word / token is going to be defined in context of the analysis to be performed (for example: any n-gram, or even any emoticon). This vector representation originates from a multinomial probability model, discussed later in this document.

A bag of words can be represented as:

$$X_i = (X_{i1}, X_{i1}, \ldots., X_{iD})$$

where each element X_{ij} is the count of word 'j' in document 'i'

And D is a vector $\in \mathbb{N}^D$

Where $\mathbb{N}$ is the set of natural numbers = {0,1,2,….}

Bag of Words Model

The model can be represented as:

$$X_i \sim \text{Multinomial}(N_i, w)$$

Where N_i is the word count in document i

and w is the weight on the j^{th} word:

$$w = (w_1, \ldots\ldots, w_D)$$

i.e: W_j is the probability of drawing word/token j from the bag (a vector of probabilities) :

$$\sum_{j=1}^{D} w_j = 1$$

and W_j >= 0 for all j

(also the object w exists in simplex space(i.e: within natural numbers))

In essence $\sum_{j=1}^{D} X_{ij}$ is equal to the total number of words in document i

What is the multinomial model?

Let's revisit the concept of the binomial model, which will help us understand the multinomial model better.

Consider a biased coin where w is the probability of getting heads, and N is the number of coin flips. Mathematically:

w = Probability(Heads)
N = # of coin flips
P(x) Where x = number of heads

Therefore :

$$P(x = k) = \binom{N}{k} w^k (1 - w)^{N-k}$$

$$\text{Where } \binom{N}{k} = \frac{N!}{k!(N-k)!}$$

A multinomial probability distribution is a generalization of the above binomial model. It generalizes the concept of the binomial model to multiple categories. It can be represented for a random variable X (Not to be confused with X_i) as:

$$X \sim \mathrm{Multinomial}(N_i, w)$$

This equation says that X has a multinomial distribution with parameter N (which represents the number of documents in the corpus) and probability vector w (probability of picking a particular word)

WHERE

$$w = (w_1,\ldots\ldots,w_D)$$

and

$$\sum_{j=1}^{D} w_j = 1$$

and w_j >= 0 for all j

On unpacking, the mathematical expression would look like

$$P(X_1 = k_1, X_2 = k_2, .., X_D = k_D) = \frac{N!}{K_1!K_2!....K_D!} * w_1^{k_1} w_2^{k_2} .... w_D^{k_D}$$

The probability that X1 is equal to k1 and x2 is equal to k2 all the way up to kd is the

ki represents the counts for the i^{th} entry in the word counts vector

Hence the multinomial probability equation for x would be:

$$\frac{N!}{K_1!\,K_2!\ldots.K_D!}\prod_{j=1}^{D} w_j^{kj}$$

and

$$\prod_{j=1}^{D} w_j^{kj}$$

is defined as

$$w_j^{k_j} = w_1^{k_1} w_2^{k_2} \ldots. w_D^{k_D}$$

Estimating w

Lets say we have a corpus of documents represented as bag of words vectors.

- $X_{1,} \ldots X_N$ where each X_i is in the set of N natural numbers in D dimensions
    - i.e: $X_i \in \mathbb{N}^{D}$
    - Which is a vector of counts
    - And we assume that

$$X_i \sim \text{Multinomial}(N_i, w)$$

The goal is to come up with an estimate $\hat{w}$, an estimate of w
An obvious estimator for this is the frequency estimator:
Which in words would be:

$$\frac{\text{The total count for word j across all docs}}{\text{Total count of all words across all documents}}$$

Mathematically this would be represented as:

$$\hat{w}_j = \frac{\sum_{i=1}^{N} X_{ij}}{\sum_{i=1}^{N} \sum_{j=1}^{D} X_{ij}}$$

Now if we assume that all documents have the same length (which is a dangerous assumption to make) the above formula can be represented as:

$$\overline{Y_j}$$

where

$$\overline{Y_j} = \frac{\sum_{i=1}^{N} Y_{ij}}{N}$$

$$\overline{Y_{ij}} = \text{frequency of word j within document i}$$

However, whenever we perform this estimation we run into two problems:

1. The frequencies can vary greatly, especially when we encounter new words. It will be hard to estimate because some will be over represented and some will be under represented.
2. The estimate can only be made on the already seen words. That is we assume that we've seen all possible words. Hence we will assign a probability of 0 to a word we haven't seen.

Smoothing : adding pseudo counts:

In a bag of words model of natural language processing and information retrieval, the data consists of the number of occurrences of each word in a document. Additive smoothing allows the assignment of non-zero probabilities to words, which do not occur in the sample.

How do we do this?

Let r be a "pseudo-count"

The smoothed estimate for the weight w_j is

$$\hat{w}_j = \frac{\sum_{i=1}^{N} \tilde{X}_{ij}}{\sum_{i=1}^{N} \sum_{j=1}^{D} \tilde{X}_{ij}}$$

where:

$$\tilde{X}_{ij} = X_{ij} + r$$

Laplace’s rule of succession allows us to apply Laplace smoothing to the above estimate. Our pseudo-count above is a smoothing constant. Under Laplacian smoothing, we will select r to be (1/N) which is the same as saying our “prior count” for all words before seeing any documents is 1.