Mike Wisniewski

CS 7322

10/5/2023

1. (32 points) Consider applying PLSI to the following corpus (each line is a separate document):

    BACCA

    CAABBA

    ACBABAB

    furthermore, assume that there are two topics, and A,B,C are the only types that are available.

    a. Now suppose we initially assigned words to topics as above (black for topic 1, red/underline for topic 2). Calculate the topic-word vectors and document-topic vectors.

Topic-Word Vectors: take the ratio of topic occurrences per document

		Topic	
		Red	Black
Document	1	0.400	0.600
	2	0.500	0.500
	3	0.429	0.571

Document-Topic Vectors: take the ratio of word occurrences per topic (corpus-wide)

		Word		
		A	B	C
Topic	Red	0.625	0.250	0.125
	Black	0.300	0.400	0.300

Sum of Probability: take the summation of topic-word vectors x document-topic vectors

		Word		
		A	B	C
Document	1	0.43	0.34	0.23
	2	0.463	0.325	0.213
	3	0.439	0.336	0.225

b. Use the vectors generated in part (a) to calculate the topic probability for each word in the corpus

Topic Probability for each word in each document: topic-word vector / (topic-word vector + document-topic vector)

			Word			
			A	B	C	sum
Document	1	Red	0.581	0.294	0.217	1.093
		Black	0.419	0.706	0.783	1.907
	2	Red	0.676	0.385	0.294	1.354
		Black	0.324	0.615	0.706	1.646
	3	Red	0.610	0.319	0.238	1.167
		Black	0.390	0.681	0.762	1.833

c. Use the result of (b) to recalculate the topic-word vectors and document-topic vectors.

Topic-Word Vectors

		Topic	
		Red	Black
Document	1	0.364	0.636
	2	0.451	0.549
	3	0.389	0.611

Document-Topic Vectors

		Word		
		A	B	C
Topic	Red	0.517	0.276	0.207
	Black	0.210	0.372	0.418

d. Calculate whether the vectors in (c) is better for the set of documents.

Sum of Probability

		Word		
		A	B	C
	1	0.322	0.337	0.341
Document	2	0.349	0.329	0.323
	3	0.329	0.335	0.336

Based on the above calculations, the distribution of probabilities across documents is more in parity than the previous iteration, therefore, this calculation produced a better set of vectors for these documents.

2. (24 points) Consider the following corpus (each line is a separate sentence):

   ABCCC

   ADBB

   CDADD

   CABB

   DACB

Suppose we want to build a bigram model based on the corpus above. Assume we have both a begin and end sentence symbol for each sentence.

Calculate the perplexity of each sentence (separately) for each of the two cases

a. The base case (no smoothing)

   Frequency

Bigram Frequency	A	B	C	D	End	
Start	2	0	2	1		5
A	0	2	1	2	0	5
B	0	2	1	0	3	6
C	1	1	2	1	1	6
D	2	1	0	1	1	5

   Bigram Model

Bigram	A	B	C	D	End
Start	0.4	0	0.4	0.2	
A	0	0.4	0.2	0.4	0
B	0	0.33	0.17	0	0.5
C	0.17	0.17	0.33	0.17	0.17
D	0.4	0.2	0	0.2	0.2

Base case Perplexity

	Prob	Perplexity
ABCCC	0.00049383	3.556893
ADBB	0.00533333	2.848395
CDADD	0.00042667	3.644617
CABB	0.00444444	2.954177
DACB	0.00133333	3.75848

b. Using Laplace (plus 1) smoothing. Note that the bigram <begin><end> will never occur so it's probability does not need to be smoothed. (Hint: you should start the smoothing process by adding 1 to the frequency of all possible bigrams)

Laplace Frequency

Bigram Frequency	A	B	C	D	End	
Start	3	1	3	2		9
A	1	3	2	3	1	10
B	1	3	2	1	4	11
C	2	2	3	2	2	11
D	3	2	1	2	2	10

Bigram Model

Bigram	A	B	C	D	End
Start	0.23	0.08	0.23	0.15	
A	0.07	0.21	0.14	0.21	0.07
B	0.07	0.20	0.13	0.07	0.27
C	0.13	0.13	0.20	0.13	0.13
D	0.21	0.14	0.07	0.14	0.14

Laplace Perplexity, using V = 4. I did not include Start and End as unique V values

	Prob	Perplexity
ABCCC	3.5165E-05	5.524783
ADBB	0.00037677	4.839334
CDADD	2.8834E-05	5.710606
CABB	0.00035165	4.906573
DACB	0.00016745	5.69144

Also show the probabilities for each bigram (preferably in a 2-d matrix).