CS7650 Problem Set 2 - Fall 2023

Due: March 31, 11:59pm ET

Please submit your solutions on Gradescope.

1 Word Embeddings

The WORD2VEC algorithm revolutionized the field of NLP by providing a high-quality, but cheaply computable means for producing continuous vector representations of words learned from a large, unlabelled corpus. Here, we will investigate the objectives used in the WORD2VEC algorithm. This question may require you to refer to Chapters 14.5, 14.6 of the Eisenstein readings.

Here is a sentence for which the algorithm will make a prediction for the missing word. The word embedding for each word in the context has been given.

Index Position	**Word**	**Embedding**
0	red	$[-2, 0, -2]$
1	green	$[2, 8, -4]$
2	and	$[2, -2, -4]$
3	?	
4	are	$[-2, -2, 2]$
5	most	$[2, 4, 2]$
6	children's	$[4, 4, 0]$
7	favorite	$[4, -2, 10]$
8	colors	$[0, 12, -16]$

Table 1: Word Embeddings for the Input Sentence.

(a) (**2 pt**) Compute the Continuous Bag-of-Words (CBOW) vector representation of the missing word for a context window h of size 3. Show your work.

(b) (**5 pt**) We've subset the vocabulary down to the words in Table (b). Fill in the scores of each word being the missing word in Table (b). Use the base-2 exponent and round to 3 decimal places.
Hint: Use dot products for this, not traditional vector-space similarity.

(c) (**1 pt**) Which word would be predicted by the CBOW algorithm to be the missing word?

Word	Embedding	Unnormalized Score	Normalized Score (P(Word))
yellow	$[-2, 4, 2]$		
pink	$[-6, 3, -6]$		
blue	$[0, 4, 2]$		
orange	$[2, 0, 0]$		
white	$[1, 3, 2]$		

Table 2: A subset of the vocabulary of the CBOW model.

2 Hidden Markov Models and the Viterbi Algorithm

We have a toy language with 2 words - "cool" and "shade". We want to tag the parts of speech in a test corpus in this toy language. There are only 2 parts of speech — NN (noun) and VB (verb) in this language. We have a corpus of text in which we the following distribution of the 2 words:

	NN	VB
cool	3	6
shade	7	4

Assume that we have an HMM model with the following transition probabilities (* is a special start of the sentence symbol).

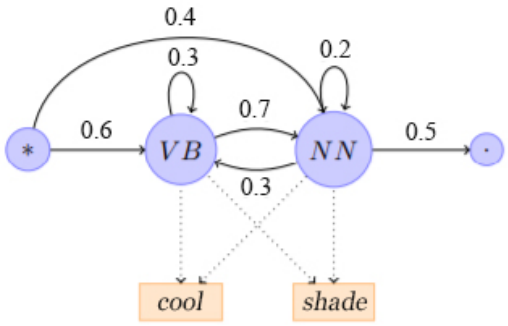

Figure 1: HMM model for POS tagging in our toy language.

1. (**2 pts**) Compute the emission probabilities for each word given each POS tag.

2. (**3 pts**) Draw the Viterbi trellis for the sequence "cool shade.". Highlight the most likely sequence. Here is an example of Viterbi trellis.

3 Named Entity Recognition

Consider a sentence that contains three named entities (organization name, person name, location name) and the predictions from four automatic name entity recognition systems. What is the entity-level Precision, Recall, and F1-score of each system's performance? Here, we do not consider giving any credits to partial matches.

Sentence	Sam	works	at	Berkshire	Hathway	headquartered	in	Nebraska
Gold Labels	B-PER	O	O	B-ORG	I-ORG	O	O	B-LOC
System #1	O	O	O	B-ORG	O	O	O	B-LOC
System #2	B-PER	O	O	O	O	O	O	B-LOC
System #3	B-PER	O	O	B-ORG	I-ORG	O	O	B-LOC
System #4	B-PER	I-PER	O	B-ORG	I-ORG	O	O	O

For each system compute:

(a) (**2 pts**) Precision

(b) (**2 pts**) Recall

(c) (**2 pts**) F-1 score

You may refer to Chapter 8.3 of the Eisenstein readings to learn more about the concept and notations used in NER.