Q2, Skip-Gram and Word2vec

Good word embeddings, such as Word2vec embeddings, often capture relational meanings. Which of the following equations do you think should hold for a good word embedding?

Notes

1. It is assumed that vector(word) gives us the embedding for the word word.
2. We use \( = \) to signal that the left hand side and right hand side values are very close to each other, although not exactly equal.
3. Korea is a country in Asia and Nigeria is a country in Africa.

(a) \( \text{vector(Nigeria)} - \text{vector(Korea)} = \text{vector(Africa)} - \text{vector(Asia)} \)
(b) \( \text{vector(Nigeria)} - \text{vector(Africa)} = \text{vector(Asia)} - \text{vector(Korea)} \)
(c) \( \text{vector(Korea)} - \text{vector(Nigeria)} = \text{vector(Africa)} - \text{vector(Asia)} \)
(d) \( \text{vector(Asia)} - \text{vector(Africa)} = \text{vector(Korea)} - \text{vector(Nigeria)} \)

The answer is (a) (d). Recall from the Vector Semantics Lecture, Slide 91, that we can use the parallelogram method to solve analogical relations of the form \( a:aa \) as \( b:bb \). Suppose we draw the embeddings for each of the words, where continents are located to the right, countries are located to the left, terms about Asia are located at the bottom, and terms about Africa are located at the top.

(a) \( \text{vector(Nigeria)} - \text{vector(Korea)} = \text{vector(Africa)} - \text{vector(Asia)} \)

(b) \( \text{vector(Nigeria)} - \text{vector(Africa)} = \text{vector(Asia)} - \text{vector(Korea)} \)
Here, we can see that the equations for (a)(d) yield vectors that point in the same direction whereas the equations for (b)(c) yield vectors that point in opposite directions. Therefore, (a)(d) is the correct answer.

Q3, Named Entity Recognition (NER) Tagging

Say we want to perform NER on the following sentence following the BIO tagging scheme.
The woman hiked up to Coit Tower in San Francisco this morning.

What tag should we assign to Francisco?

(a) B-LOC  
(b) I-LOC  
(c) O  
(d) None of the above

The answer is (b). Francisco is part of the named entity San Francisco, which is a location. It is not the first token in the entity, so it does not get the tag B-LOC, but instead gets the tag I-LOC.

Q4, Skip-Gram and Word2vec

Assume a +/- 2 word window, given the training sentence: The quick brown fox jumped over the lazy dog.

Suppose we are trying to learn the embedding for our target fox. Which of the following is/are true about our method?

Select all that apply.

(a) We are trying to maximize the similarity to a pair like (fox, brown)  
(b) We are trying to maximize the similarity to a pair like (fox, dog)  
(c) We are trying to minimize the similarity to a pair like (fox, octopus)  
(d) We are trying to minimize the similarity to a pair like (fox, quick)

The answer is (a) (c). We are instructed to assume a +/- 2 word window with the target fox. This gives us a training sentence which looks like The [quick brown fox jumped over] the lazy dog where quick, brown, jumped, and over are all included in the context for our target fox.

Recall from the Vector Semantics Lecture, Slide 79, that the goal of learning in Word2vec is to maximize the similarity of target word, context word pairs \(w, c_{pos}\) drawn from the positive data and to minimize the similarity of \(w, c_{neg}\) pairs drawn from the negative data. This means that since brown and quick are included in our context, we want to maximize the similarity to pairs like (fox, brown) and (fox, quick). However, since dog and octopus are not included in our context, we want to minimize the similarity to pairs like (fox, dog) and (fox, octopus).

Therefore (a)(c) is the correct answer.
• (Q5, Computation Graphs and Backpropogation)

Given the following graph, what is the minimum number of partial derivatives you need to calculate in order to find $\frac{\partial L}{\partial c}$?

(a) 2  
(b) 3  
(c) 4  
(d) 5  
(e) 6

The answer is (b). Using the chain rule, we can calculate $\frac{\partial L}{\partial c} = \frac{\partial L}{\partial g} \ast \frac{\partial g}{\partial f} \ast \frac{\partial f}{\partial c}$. So, we need 3 derivatives to find $\frac{\partial L}{\partial c}$.

Q6: Softmax and Sigmoid

Which of the following statements are correct? Select all that apply.

(a) The Sigmoid function takes a vector of scores and produces a vector of probabilities.  
(b) The Sigmoid function takes a vector of scores and produces a single scalar which is a probability.  
(c) The Softmax function takes a vector of scores and produces a vector of probabilities.  
(d) The scores input to Softmax and Sigmoid functions can be both positive and negative.

The answer is (c) (d) or (a) (c) (d).

The definition of the sigmoid function is (see slide 5 of Week 5’s slides on neural networks):

$$y = \sigma(z) = \frac{1}{1 + e^{-z}}$$

where both the input $z$ and output $y$ are scalars; therefore (b) is incorrect.

However, some people may have interpreted selection (a) referring to the element-wise version of the sigmoid function, where we apply the sigmoid function independently to each element of a
vector of scores, and obtaining a vector of probabilities. Under this interpretation (a) is considered correct; hence we gave credit for both combinations (c) (d) or (a) (c) (d).

The definition of the softmax function is (see slide 31 of Week 5’s slides on neural networks):

Given a vector \( \mathbf{z} \) with \( k \) elements:

\[
\text{softmax}(\mathbf{z}) = \left( \frac{e^{-z_1}}{\sum_{j=1}^{k} e^{-z_j}}, \frac{e^{-z_2}}{\sum_{j=1}^{k} e^{-z_j}}, \ldots, \frac{e^{-z_k}}{\sum_{j=1}^{k} e^{-z_j}} \right)
\]

where the output of the softmax function is also a vector with \( k \) elements. Notice the recurring term \( \sum_{j=1}^{k} e^{-z_j} \) on the denominator of each element. That guarantees that all of the elements in \( \text{softmax}(\mathbf{z}) \) sum up to 1, so it is a vector of probabilities. Therefore (c) is correct.

From the graph of the sigmoid function, we see that its inputs can be any real number. Same for the inputs of the softmax function. Only their outputs are bound within the range of 0 to 1. Hence (d) is correct.

**Q7 IR for Chatbots**

The answer is B: \( r_2 \)

Use cosine similarity to calculate each pair of \( q \) with each : \( r \). The vector for : \( r_2 \)

**Q8 Dialogue-state Architecture**

The answer is C “Given a dialogue act and slots and values, generate a natural language utterance.”

See slide 104 on in the Chatbot Slides

**Q9 Content-based vs. Collaboration Filtering**

The answer is B: Content-based recommender systems can more easily recommend new or niche items to users compared to purely collaborative filtering-based recommender systems.

People are more complex than content and generally harder to predict. Thus, in practice, content-based recommender systems tend to have better results (as in the slides)

**Q10 Item-item Similarity**

The answer is B: The Octopus Game

To calculate, mean-centered overlapping-item cosine similarity:

\[ S_{xy} = \text{items rated by both users } x \text{ and } y \]
\[
sim(x, y) = \frac{\sum_{s \in S_{xy}} (r_{xs} - \bar{r}_x)(r_{ys} - \bar{r}_y)}{\sqrt{\sum_{s \in S_{xy}} (r_{xs} - \bar{r}_x)^2} \sqrt{\sum_{s \in S_{xy}} (r_{ys} - \bar{r}_y)^2}}
\]

This is the same as Mean-centered cosine similarity but it excludes items that aren’t rated by both users. First mean center all the vectors. Then, calculate mean-centered cosine similarity between each row and Todd Lasso.

**Q11 Ethical Implications**

Answer is D: they create an informational barrier that prevents people from being exposed to differing viewpoints.

**Q12 Transition Probability Matrix**

The answer is 0.44. To compute \( P \), we need to first calculate the transition probabilities from each node to its neighbors. For a given node \( i \), if it has \( k \) neighbors, then the probability of transitioning to each neighbor is \( 1/k \). If \( i \) has no neighbors, then it transitions to all other nodes with equal probability. Then we can complete the power iteration method to repeatedly multiply the matrix until the product looks stable.

**Q13 PageRank**

The correct answers are “dead ends” and “isolated nodes” (nodes without any incoming or outgoing edges).

Teleporting is an essential component of the PageRank algorithm. It allows the user to randomly jump to any webpage, regardless of the links, and start again. This ensures that the algorithm converges and that no webpages have a PageRank of zero.

If we do not allow teleporting, the PageRank algorithm will not converge, and the results will not be valid. In particular, link structures with dead ends and isolated nodes will invalidate PageRank.

Dead ends are nodes with no outgoing edges. If the user ends up on a dead-end page, they cannot continue to another page, and the algorithm will not converge.

Isolated nodes are nodes with no incoming or outgoing edges. These nodes cannot contribute to the PageRank of any other node, and their PageRank will always be zero.

Therefore, if we do not allow teleporting, the PageRank algorithm will be invalidated for link structures with dead ends and isolated nodes.
Q14 Strong Triadic Closure
Answer: The shortest path from Alexis to Pablo passes through a strong link

Q15 Strong Triadic Closure
Answer: Directed edges, maximum in-degree = 1