1. Consider the following corpus: "human machine interface for computer applications. user opinion of computer system response time. user interface management system. system engineering for improved response time". What is the size of the vocabulary of the above corpus ?

   A. 13

   B. 15

   C. 14

   D. 16

   Solution: The vocabulary of the above corpus will be as follows:
   [human, machine, interface, for, computer, applications, user, opinion, of, system, response, time, management, engineering, improved].
   The length or size of the vocabulary is 15. **Option B** is the correct answer.

2. Let $count(w, c)$ be the number of times the words w and c appear together in the corpus (*i.e.*, occur within a window of few words around each other). Further, let $count(w)$ and $count(c)$ be the total number of times the word w and c appear in the corpus respectively and let N be the total number of words in the corpus. The PMI between w and c is then given by:

   A. $\log \frac{count(w,c)*count(w)}{N*count(c)}$

   B. $\log \frac{count(w,c)*count(c)}{N*count(w)}$

   C. $\log \frac{count(w,c)*N}{count(w)*count(c)}$

   D. $\log \frac{count(w)*count(c)}{count(w,c)*N}$

   Solution: Option C is the correct answer.

3. The SVD of a matrix X is given by $X = U\Sigma V^\top$ where U contains the eigen vectors of:

   A. X

   B. $XX^\top$

   C. $X^\top$

   D. $X^\top X$

Solution: This follows from the SVD theorem. X can be written as $X = U\Sigma V^\top$ where U contains the eigen vectors of $XX^\top$, V contains the eigen vectors of $X^\top X$. Therefore, **Option B** is the correct answer.

4. Let X be the co-occurrence matrix such that the (i, j)-th entry of X captures the PMI between the i-th and j-th word in the corpus. Every row of X corresponds to the representation of the i-th word in the corpus. Suppose each row of X is normalized (*i.e.*, the L_2 norm of each row is 1) then the (i, j)-th entry of $XX^\top$ captures the:

   A. PMI between word i and word j

   B. euclidean distance between word i and word j

   C. probability that word i co-occurs with word j

   D. cosine similarity between word i and word j

Solution: Option D is the correct answer.

5. Let the co-occurrence matrix $X \in \mathbb{R}^{m \times n}$ (*i.e.*, there are m words and n context words). Once we do a k-rank approximation of X using SVD, we take $W_{word} = U\Sigma$ as the matrix containing the representations of the words. What are the dimensions of W_{word} ?

   A. $m \times n$

   B. $n \times k$

   C. $m \times k$

   D. $k \times m$

Solution: Option C is the correct answer

6. At the input layer of continuous bag of words model, we multiply a one-hot vector $x \in \mathbb{R}^{|V|}$ with the parameter matrix $\mathbf{W} \in \mathbb{R}^{k \times |V|}$. What does each column of $\mathbf{W}$ correspond to ?

   A. the representation of the i-th word in the vocabulary

   B. the i-th eigen vector of the co-occurrence matrix

Solution: Option A is the correct answer

7. Consider the word w and a word c which appears before it. For example, w could be the word *barks* and c could be the word *dog*. Let v_w and u_c be the representations of w and c respectively. Further, assume that you are training the bag-of-words model using $n = 1$ (*i.e.*, you are training the model to predict the next word given the current word). The loss function used in the continuous bag-of-words model ensures that:

   A. v_w and u_c are orthogonal to each other

   B. v_w and u_c are similar to each other

   C. does not guarantee anything about v_w and u_c (after all, in the above example, why should the algorithm care about the relation between the representations of *dog* and *barks*. It should rather be interested in the relation between the representations of {*dog* and *cat*} or {*barks* and *howls*})

Solution: Option B is the correct answer

8. Consider the word w and a word c which appears before it. For example, w could be the word *barks* and c could be the word *dog*. Let v_w and u_c be the representations of w and c respectively. Further, assume that you are training the bag-of-words model using $n = 1$ (*i.e.*, you are training the model to predict the next word given the current word). Let $\hat{y}$ be the output of the model (*i.e.*, $\hat{y}$ is the probability distribution over all words in the vocabulary). In particular, $\hat{y}_w$ is the probability assigned by the model to the word w. If you are using gradient descent to train the model and η is the learning rate then the update rule for v_w is given by:

   A. $v_w = v_w + \eta\ u_c(1 - \hat{y}_w)$

   B. $v_w = v_w + \eta\ \hat{y}_w(1 - u_c)$

   C. $v_w = v_w - \eta\ u_c(1 - \hat{y}_w)$

   D. $v_w = v_w - \eta\ \hat{y}_w(1 - u_c)$

Solution: Option A is the correct answer