Questions on Chapter 3

1. List the four categories of text representation techniques.?

text representation : We're given a piece of text, and we're asked to find a scheme to represent it mathematically.

These approaches are classified into four categories:

Basic vectorization approaches
Distributed representations
Universal language representation
Handcrafted features

2. Describe the concept vector space models.?

in order for ML algorithms to work with text data, the text data must be converted into some mathematical form. represent text units (characters, phonemes, words, phrases, sentences, paragraphs, and documents) with vectors of numbers. This is known as the vector space model (VSM)

VSM -> It's a mathematical model that represents text units as vectors

3. Use "D1: Dog bites man, D2: Man bites dog, D3: Dog eats meat, and D4: Man eats food" as an input, find their representation using one-hot encoding, bag of words, bag of N-gram, and TF-IDF.?

one-hot encoding :

In one-hot encoding, each word w in the corpus vocabulary is given a unique integer ID word id that is between 1 and |V|, where V is the set of the corpus vocabulary. Each word is then represented by a V-dimensional binary vector of 0s and 1s.

On the positive side, one-hot encoding is intuitive to understand and straightforward to implement.

However, it suffers from a few shortcomings:
• The size of a one-hot vector is directly proportional to size of the vocabulary, and most real-world corpora have large vocabularies
• This representation does not give a fixed-length representation for text
if a text has 10 words, you get a longer representation for it as compared to a text with 5 words. For most learning algorithms, we need the feature vectors to be of the same length.

• Say we train a model using our toy corpus. At runtime, we get a sentence: “man eats fruits.” The training data didn’t include “fruit” and there’s no way to represent it in our model. This is known as the out of vocabulary (OOV) problem.

BOW :

represent the text under consideration as a bag (collection) of words while ignoring the order and context

the advantages of this encoding:
• Like one-hot encoding, BoW is fairly simple to understand and implement.
• With this representation, documents having the same words will have their vector representations closer to each other
• We have a fixed-length encoding for any sentence of arbitrary length.

However, it has its share of disadvantages, too:
• The size of the vector increases with the size of the vocabulary.
• It does not capture the similarity between different words that mean the same thing. Say we have three documents: “I run”, “I ran”, and “I ate”. BoW vectors of all three documents will be equally apart.
• This representation does not have any way to handle out of vocabulary words
• As the name indicates, it is a “bag” of words—word order information is lost in this representation

Bag of N-Grams :
It does so by breaking text into chunks of n contiguous words (or tokens).This can help us capture some context, which earlier approaches could not do.

Each chunk is called an n-gram. The corpus vocabulary, V, is then nothing but a collection of all unique n-grams across the text corpus. Then, each document in the corpus is represented by a vector of length |V|

Here are the main pros and cons of BoN:
• It captures some context and word-order information in the form of n-grams.
• Thus, resulting vector space is able to capture some semantic similarity. Documents having the same n-grams will have their vectors closer to each other
• As n increases, dimensionality (and therefore sparsity) only increases rapidly.
• It still provides no way to address the OOV problem.

TF-IDF :
it aims to quantify the importance of a given word relative to other words in the document and in the corpus.

If we look back at all the representation schemes we've discussed so far, we notice three fundamental drawbacks:

• They're discrete representations , they treat language units (words, n-grams, etc.) as atomic units. This discreteness hampers their ability to capture relationships between words.

• The feature vectors are high-dimensional representations.
• They cannot handle OOV words

4. Explain the difference between (a) distributional similarity and distributional hypothesis (b) distributional representation and distributed representation.?

distributional similarity -> This is the idea that the meaning of a word can be understood from the context in which the word appears. This is also known as connotation: meaning is defined by context

distributional hypothesis -> hypothesizes that words that occur in similar contexts have similar meanings.

distributional representation -> This refers to representation schemes that are obtained based on distribution of words from the context in which the words appear

distributed representation -> the vectors in distributional representation are very high dimensional. This makes them computationally inefficient and hampers learning. To alleviate this, distributed representation schemes significantly compress the dimensionality

5. Describe the wording embedding concept with an example of its use. ?

A word embedding is a learned representation for text where words that have the same meaning have a similar representation.

Example -> If we're given the word "USA," distributionally similar words could be other countries (e.g., Canada, Germany, India, etc.) or cities in the USA. If we're given the word "beautiful," words that share some relationship with this word (e.g., synonyms, antonyms) could be considered distributionally similar words. These are words that are likely to occur in similar contexts.

"Word2vec," based on "distributional similarity," can capture word analogy relationships

6. Explain with an example the two architectural variants of Word2vec: CBOW and SkipGram.?

CBOW : In CBOW, the primary task is to build a language model that correctly predicts the center word given the context

Given a sentence of, say, m words, it assigns a probability Pr(w1 , w2 , ….., wn) to the whole sentence. The objective of a language model is to assign probabilities in such a way that it gives high probability to "good" sentences and low probabilities to "bad" sentences. By good, we mean sentences that are semantically and syntactically correct. By bad, we mean sentences that are incorrect—semantically or syntactically or both. So, for a sentence like "The cat jumped over the dog," it will try to assign a probability close to 1.0, whereas for a sentence like "jumped over the the cat dog," it tries to assign a probability close to 0.0.

SkipGram : SkipGram is very similar to CBOW, with some minor changes. In Skip- Gram, the task is to predict the context words from the center word.

7. How the OOV problem can be solved?

- create vectors that are initialized randomly, where each component is between –0.25 to +0.25
- There are also other approaches that handle the OOV problem by modifying the training process by bringing in characters and other subword-level
- can handle the OOV problem by using subword information, such as morphological properties (e.g., prefixes, suffixes, word endings, etc.), or by using character representations.

8. What is the difference between Doc2vec and Word2vec?

Word2vec learned representations for words, and we aggregated them to form text representations.

fastText learned representations for character n-grams.

Doc2vec, which allows us to directly learn the representations for texts of arbitrary lengths (phrases, sentences, paragraphs, and documents) by taking the context of words in the text into account.

9. What are the important aspects to keep in mind while using word embeddings?

- All text representations are inherently biased based on what they saw in training data.
- We still need ways to encode specific aspects of text, the relationships between sentences in it
- pre-trained embeddings are generally largesized files (several gigabytes), which may pose problems in certain deployment scenarios.

10. How high-dimensional data can be represented visually?

t-distributed Stochastic Neighboring Embedding. It's a technique used for visualizing high-dimensional data like embeddings by reducing them to two or three-dimensional data.

11. With example explain the use of handcrafted feature representations?

TextEvaluator, It's software developed by Educational Testing Service. The goal of this tool is to help teachers and educators provide support in choosing grade-appropriate reading materials for students and identifying sources of comprehension difficulty in texts.

measures such as "syntactic complexity" and "concreteness" etc., cannot be calculated by only converting text into BoW or embedding representations. They have to be designed manually, keeping in mind both the domain knowledge and the ML algorithms to train the NLP models. This is why we call these handcrafted feature representations.