1 . **What types of tokenizers do you know? Compare them**

Ans - 1) White Space 2) Word Tokenaizer 3) Sentence Token. 4) Subword 5) Character 6) Custom

2. **How do regular tokens differ from special tokens?**

Ans - **Regular** = individual elements like keywords, identifiers, operators, and literals

Special Tokens = parentheses, brackets, or commas, which have particular roles in syntax or structure

3. **What tokenizer is used in BERT, and which one in GPT?**

Ans - BERT - WordPiece tokenization

GPT - Byte Pair Encoding (BPE) tokenization

4.What is normalization in TF-IDF

Ans. Normalization in TF-IDF (Term Frequency-Inverse Document Frequency) refers to the process of scaling the raw term frequencies and inverse document frequencies to ensure that the resulting TF-IDF values are comparable across different terms and documents.

5. Explain possible methods for text preprocessing (lemmatization and stemming). What algorithms do you know for this, and in what cases would you use them?

Ans.

6. What metrics for text similarity do you know?

Ans - Cosine Similarity, TF-IDF, WordEmbedding

7. What are vanishing gradients for RNN? How do you solve this problem?

In recurrent neural networks (RNNs), vanishing gradients refer to a situation where the gradients computed during backpropagation become very small as they are propagated back through time steps. This phenomenon can occur due to the nature of the RNN architecture and the use of certain activation functions, such as the sigmoid or hyperbolic tangent (tanh) functions.

When the gradients become very small, they effectively "vanish," meaning they don't contribute significantly to updating the parameters (weights) of the network during training. As a result, the model may struggle to learn long-range dependencies and may not effectively capture sequential patterns in the data.

1. ****What is the difference between stemming and lemmatization?****
   - *Answer:* Stemming reduces words to their root form by removing suffixes, while lemmatization reduces words to their base or dictionary form (lemma), considering the word's meaning and context.

2. ****Explain the concept of tokenization in NLP**.****
   - *Answer:* Tokenization is the process of breaking down text into smaller units called tokens, which can be words, subwords, or characters. These tokens serve as the basic units of input for NLP tasks.

3. ****What is the purpose of the attention mechanism in transformers?****
   - *Answer:* The attention mechanism in transformers enables the model to focus on relevant parts of the input sequence during processing. It assigns weights to different tokens, allowing the model to weigh the importance of each token when making predictions.

4. ****Describe how recurrent neural networks (RNNs) handle sequential data**.****
   - *Answer:* RNNs process sequential data by maintaining a hidden state that captures information from previous time steps. At each time step, the model updates the hidden state based on the current input and the previous hidden state, allowing it to capture temporal dependencies in the data.

6. ****Explain the difference between word embeddings and character embeddings.****
   - *Answer:* Word embeddings represent words as dense vectors in a continuous vector space, capturing semantic relationships between words. Character embeddings, on the other hand, represent words as sequences of character embeddings, capturing morphological and orthographic information.

9. ****What are some common evaluation metrics used for text summarization tasks?****
   - *Answer:* Common evaluation metrics for text summarization include ROUGE (Recall-Oriented Understudy for Gisting Evaluation), BLEU (Bilingual Evaluation Understudy), METEOR (Metric for Evaluation of Translation with Explicit Ordering), and CIDEr (Consensus-based Image Description Evaluation).

10. ****Explain how self-attention works in transformer models**.****
    - *Answer:* Self-attention allows transformer models to weigh the importance of different words in the input sequence when making predictions. It computes attention scores between all pairs of words in the sequence, applies softmax to obtain attention weights, and then combines the input embeddings with the attention weights to generate context-aware representations.

These questions cover a range of topics in NLP, from fundamental concepts like tokenization and embeddings to more advanced topics like attention mechanisms and machine translation challenges.