NLP Interview Qs

Basics of NLP:

1. What is Natural Language Processing (NLP), and what are its applications?

2. Explain the difference between text classification and named entity recognition.

3. How does sentiment analysis work, and why is it useful?

4. Describe a scenario where machine translation could be valuable.

5. What are the challenges of processing human languages compared to structured data?
    - Ambiguity: الكلمة ليها أكتر من معنى
      - Named Entity Recognition + Context analysis.
    - Context Dependency.
      - Attention Mechanisms
    - Sarcasm:
      - Sentiment analysis.
    - Idioms.

    - Non-standard grammar and spelling:
      - Tokenization and Normalization.
    - Out-of-Voicabulary words:
      - Character Level Model.

Text Preprocessing:

6. What is tokenization, and why is it important in NLP?

7. How does stopword removal contribute to text processing?

8. What is the difference between stemming and lemmatization?

9. Why might lemmatization be preferred over stemming in certain cases?

10. How would you handle special characters and punctuation during text preprocessing?

Word Embeddings:

11. Explain the concept of word embeddings and their purpose.

    capture the semantic relationships and contextual meanings of words based on their usage in a large corpus of text.

12. Compare Word2Vec and GloVe algorithms for generating word embeddings.

    W2V: predict a word within its context (Continuous Bag of Words)

    Predicting context words gives a target word.

    Word2Vec uses a neural network to maximize the likelihood of the correct context words given a target word.

    The embeddings are learned through backpropagation, adjusting the weights of the neural network to minimize the prediction error.

    Word2Vec is more known for its analogy-solving capabilities, while **GloVe** is better at handling global semantic information.

    → Captures the semantic relationships and contextual information between words.

13. How do word embeddings capture semantic relationships between words?

14. What is the "word analogy" task, and how can word embeddings solve it?
king - man + woman = queen

15. Describe the concept of "distributional similarity" in word embeddings.
Words with similar meanings tend to occur in similar contexts.

Neural Networks for NLP:

16. What are Recurrent Neural Networks (RNNs), and where are they commonly used in NLP?
- Handle sequential data. - Order of elements matters.
- Key Characteristics of RNNs:
  A. Recurrent Connections (time step + hidden state)
  B. Vanishing gradient,
  C. Parameter sharing,

17. Explain the vanishing gradient problem in RNNs and how it affects learning long-range dependencies.
- Due to Backpropagation and long sequences while Gradient Descent updates the weights and bias based on the error gradient.
When gradients become extremely small (close to zero) during backpropagation, they cause the network's weights to be updated very slowly or not at all.
- Gradient Descent: update weights and bais: old + Learning rate * div(loss(actual - predicted))
- Backpropagation:
  Forward pass
  Backward pass
  Chain Rule
Gradient of the loss: the derivatives of the loss function with respect to the parameters of a model.
In summary, backpropagation is the process of calculating gradients using the chain rule, and these gradients are then used in the gradient descent algorithm to update the parameters of the neural network. The chain rule helps break down

the gradient computation across the layers, and the gradient descent algorithm guides the parameter updates to minimize the loss function.

18. How do Long Short-Term Memory (LSTM) networks address the vanishing gradient problem?
    1. Memory Cell
    2. Gating Mechanism:
    Forget: which information from the previous cell state to discard and what new information to store.,
    Input: This gate determines which new information should be stored in the cell state. It involves a sigmoid layer that decides which values to update and a tanh layer that generates new candidate values.
    Output Gate: This gate controls the information that will be output from the cell state. It considers the cell's current state and filters the information to produce the final output.
    Skip Connections: residual connection.
    They involve creating shortcuts that allow the output of one layer to be added to the output of one or more previous layers.

19. What is the purpose of an attention mechanism in sequence-to-sequence tasks?

20. Describe the differences between encoder and decoder architectures in sequence-to-sequence models.

Transformers:

21. What are Transformers, and why are they considered a breakthrough in NLP?
22. Explain the self-attention mechanism in Transformers.
    Key, Query, Value.
    The key/value/query concept is analogous to retrieval systems. For example, when you search for videos on Youtube, the search engine will map your **query** (text in the search bar) against a set of **keys** (video title, description, etc.) associated with candidate videos in their database, then present you the best matched videos (**values**).

1. Calculate Q, K, V 2. Calculate Attention Scores: Dot Product of Q, K of other words..
3. Apply softmax and weighted sum.

23. How do Transformers process sequences in parallel compared to RNNs?
    Long-Range Dependencies: Self-attention enables transformers to capture long-range dependencies in sequences, as any word can potentially influence any other word's representation.
    Non-Sequential Processing: Unlike traditional recurrent models, transformers can process tokens in parallel due to self-attention, making them highly efficient for sequence tasks.
    Task Adaptation: Transformers can be fine-tuned for various NLP tasks by stacking encoder and decoder layers, adjusting the self-attention mechanism, and adding task-specific components.
24. What is the role of positional encodings in Transformer models?
    Transformers use positional encodings to provide information about the position of words in the sequence. These encodings are added to the word embeddings and do not involve sequential processing.
25. Describe the components of an encoder and a decoder in a Transformer architecture.

BERT and Pretrained Models:

Contextual Understanding: BERT learns rich contextual representations that are essential for understanding nuances in language, capturing word meanings that depend on the surrounding context.
Transfer Learning: The pre-trained BERT model can be fine-tuned on downstream NLP tasks, such as text classification, named entity recognition, and question answering. The contextual embeddings learned during pretraining provide a strong foundation for these tasks.
The decoder in a transformer-based architecture is typically used for tasks like sequence generation, language translation, and text generation, where the output is generated one token at a time based on the context representation created by the encoder. In such tasks, the decoder takes the context representation and generates an output sequence step by step.

Models that use both encoder and decoder components are often referred to as "sequence-to-sequence" models, where the encoder processes the input sequence to create a context representation, and the decoder generates the output sequence based on that context.

If you're interested in models that use the decoder-only architecture, you might want to explore models like GPT (Generative Pretrained Transformer) series, where the model is designed for autoregressive language generation. GPT models, such as GPT-2 and GPT-3, utilize decoder-only architectures and generate text one token at a time based on the context from preceding tokens.

To summarize, BERT is an encoder-based model designed for bidirectional context modeling and pretraining tasks. For decoder-based architectures, you might want to look into models like GPT, which focus on autoregressive text generation.

Sequence-to-Sequence Models:

31. What is a sequence-to-sequence model, and in what tasks are they commonly used?
32. Explain how attention mechanisms improve the performance of sequence-to-sequence models.
33. How does beam search differ from greedy decoding in sequence generation tasks?
34. Describe a scenario where a sequence-to-sequence model could be used for text generation.
35. What are some challenges when training sequence-to-sequence models on long sequences?

Evaluation Metrics:

36. Define accuracy, precision, recall, and F1-score in the context of NLP evaluation.
37. Explain the BLEU score and its purpose in evaluating machine translation systems.
38. What does the ROUGE score measure, and in which task is it commonly used?
39. How can you handle cases where reference summaries differ in length during ROUGE calculation?

40. Describe a situation where a high BLEU score might not necessarily indicate good translation quality.

NLP Libraries and Tools:

41. What is NLTK, and how is it used in NLP tasks?
42. Describe the features and capabilities of the spaCy library.
43. How does the Hugging Face Transformers library facilitate working with pre-trained language models?
44. Explain the concept of a tokenizer in the context of the Transformers library.

NLP Challenges:

46. How would you address the issue of bias in NLP models?
47. Describe a scenario where a model trained on one domain doesn't perform well in another.
48. What are out-of-vocabulary words, and how can they impact NLP models?
49. How could you handle the challenge of translating idiomatic expressions or cultural references?
50. Discuss the ethical considerations when using NLP for sentiment analysis of social media posts.

General Qs in NLP:

51. What is the purpose of the term frequency-inverse document frequency (TF-IDF) technique in text analysis?
52. Explain the concept of n-grams and their significance in language modeling.
53. How can you handle spelling errors and typos in text data before analysis?
54. What is topic modeling, and how can it be used to uncover latent themes in a collection of documents?
55. Describe the difference between rule-based and machine learning-based named entity recognition.

56. How do you handle text data that contains multiple languages or code-switching?
57. Explain the concept of word sense disambiguation and its importance in NLP.
58. What are language models, and how have advancements in pre-trained models impacted NLP tasks?
59. Describe the process of word alignment in statistical machine translation.
60. How can you evaluate the readability of a piece of text, and why is it important?
61. Explain the concept of cross-lingual embeddings and how they can facilitate multilingual NLP.
62. What is sentiment lexicon, and how is it used in sentiment analysis?
63. How can you address the challenge of sarcasm detection in NLP models?
64. Describe the concept of named entity linking (NEL) and its role in knowledge extraction.
65. Explain the concept of co-reference resolution and its significance in natural language understanding.
66. What is the difference between unsupervised, supervised, and semi-supervised learning in NLP?
67. How can you address the challenge of data privacy when working with text data containing personal information?
68. Describe the process of training a word2vec model and how the resulting embeddings capture semantic relationships.
69. What is text augmentation, and how can it improve the performance of NLP models?
70. Explain the concept of word entropy and how it can be used to quantify the uncertainty of word prediction in language models.
71. How do you assess the generalization ability of an NLP model to ensure it performs well on new, unseen data?
72. Describe the concept of zero-shot learning and how it is applied in NLP.
73. What is the purpose of attention visualization in transformer models, and how can it provide insights into model behavior?
74. How can you handle imbalanced datasets in sentiment analysis tasks?
75. Explain the concept of context window in word embeddings and how it influences the quality of embeddings.

76. What is a chatbot, and how does it differ from a traditional software application?
77. Describe the components of a chatbot architecture.
78. How can rule-based chatbots be designed, and what are their limitations?

79. **Explain the concept of intent recognition in chatbots.**
Intent recognition involves identifying the user's intention or purpose behind their

input. It's a key step in understanding what the user wants and generating relevant responses. Intent recognition is often achieved through machine learning models like intent classifiers, which are trained on labeled data to associate user input with specific intentions or goals.

80. **What are entities in the context of chatbot interactions?**
    Entities are specific pieces of information within user input that provide context for generating accurate responses. For example, in the query "Book a flight from New York to Los Angeles," "New York" and "Los Angeles" are entities representing departure and arrival locations. Entity recognition involves identifying and extracting such information from the user's text.

81. **Compare and contrast generative and retrieval-based chatbots.**
    Generative Chatbots: These models generate responses from scratch using language generation techniques, often based on large language models. They can produce novel responses but may lack control over output.
    Retrieval-Based Chatbots: These systems select responses from a predefined set of responses or templates based on the context of the conversation. They provide more control over responses but might be less creative.

82. **How can you evaluate the performance of a chatbot's responses?**
83. **What is transfer learning, and how is it used in training chatbots?**
84. **Describe the concept of reinforcement learning in training chatbots.**

85. **How can you handle out-of-scope queries in a chatbot effectively?**

86. **Explain the concept of slot filling and its role in maintaining context during a conversation.**

87. **What are multi-turn conversations, and how can chatbots handle them?**

88. **How do you implement sentiment analysis in chatbots to tailor responses based on user sentiment?**

89. **What is open-domain vs. closed-domain chatbot, and what are their respective use cases?**

90. **How can you make a chatbot sound more human-like and engaging in conversations?**

91. **Describe the challenges of handling ambiguous user input in chatbot interactions.**

92. **What is the importance of context in chatbot conversations, and how can it be maintained effectively?**

93. **Explain the role of pre-trained language models in building chatbots.**

94. **How can you integrate a chatbot into various platforms and communication channels?**

95. **What are some ethical considerations when designing and deploying chatbots, especially in sensitive domains?**