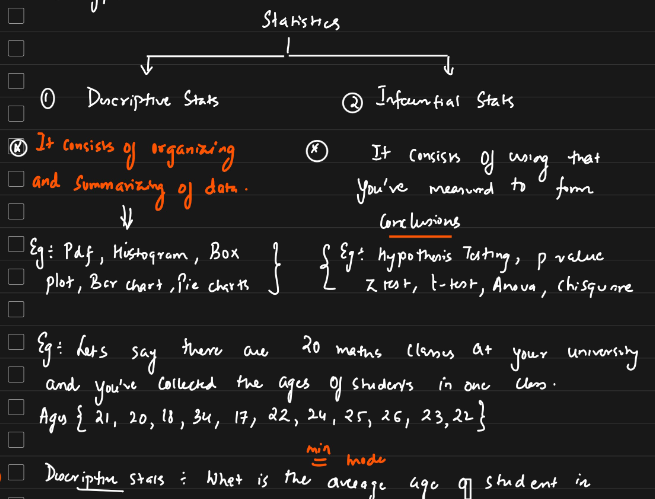
Gen AI Specialist Interview Questions:

Topics:

* Python
* Statistics
* Machine Learning - NLP
* Deep Learning (NLP)

Statistics:

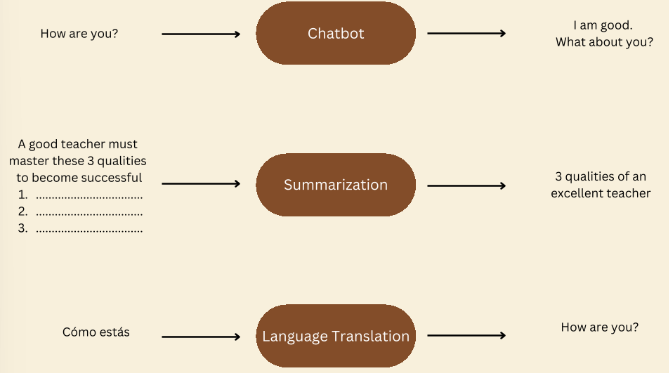
* What is statistic?
* What is population and sample?
* What is central tendency? ( mean, median and mode)
* Difference between descriptive statistics and Inferential statistics - Hypothesis testing (z test, t test, chi square, ANOVA)  
  
* How these tests are used in the real-world scenarios?
* What is the measure of dispersion? Variance, standard deviation
* What is sampling method and types?
* What is histogram?
* What is inter quartile range?
* What are boxplots and what information will you get?
* What is the effect of outliers and how to remove it?
* What is normal distribution and gaussian distribution?
* What is z score and when to use it?
* What is the difference between standardization and normalization?
* What is central limit theorem?
* What is covariance?
* What is correlation? Types? Examples?
* What is pearson correlation coefficient?
* What is T test?
* What is annova test?
* What is chisquare test?
* What is one tail and q tail test?
* What is hypothesis testing and p value?
* What is confidence interval in statistics?
* What is type1 and type 2 error?

Machine learning:

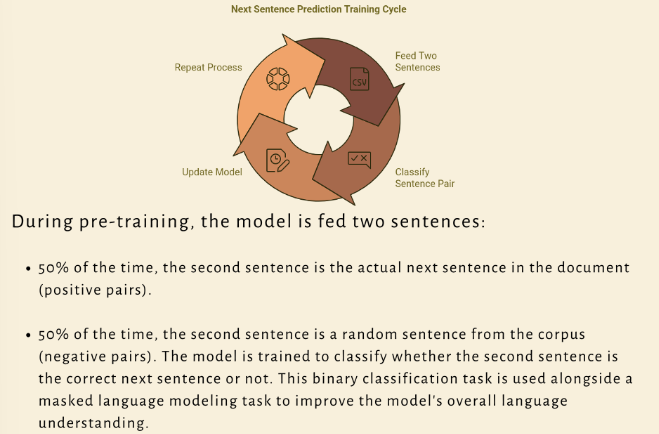
* Implementation - how dataset looks like, how are you going to prepare your dataset, what will be the input and outputs, what is vector size
* Cosine similarity score
* Simple Linear Regression

Deep Learning:

1. Activation function
2. Loss Function
3. Optimizer
4. What is context window?
5. What is hyperparameter?
6. What is sequence to sequence models?   
   Type of neural network architecture designed to transform one sequence of data into another sequence. These models are commonly used in tasks where the input and output have variable lengths, such as in machine translation, text summarization, and speech recognition.



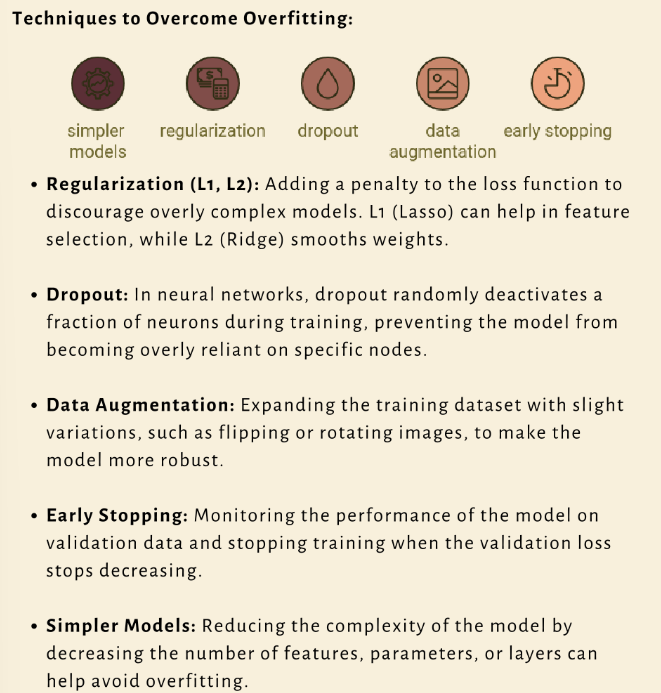
1. What is next sentence prediction and how is useful in language modelling?   
   Used in training large models like BERT (Bidirectional encoder representations from transformers). Helps a model understand the relationship between two sentences, which is important for tasks like question answering, dialogue generations, and information retrieval.



1. What is overfitting in ML, and how can it be prevented?   
   Occurs when ML model performs well on the training data but poorly on unseen of test data. This happens because the model has learned not only the underlying patterns in the data but also the noise and outliers, making it overly complex and tailored to the training set. As a result, the model fails to generalize to new data.

Techniques to overcome overfitting:

Regularization, dropout, data augmentation, early stopping, simpler models.

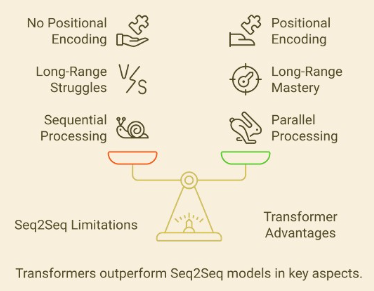


* Transformers - what is the architecture, what is the encoder and decoder, what is attention> (refer to attention is all you need paper), what is self attention, etc.
* BERT - How can you use hugging face and implement this?

NLP Interview questions

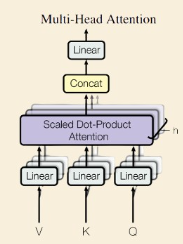
1. What is Tokenisation? Tell me some application where it is being used?
2. What is text preprocessing? Tokenization, lowercase, regular expression, remove punctuation, digits, stop words, replace specific words, remove urls, htmls tags, extra white spaces etc.
3. What is text normalization? Stemming and lemmatization.
4. What is difference between stemming and lemmatization?
5. What is stemming and provide an example? Process of reducing infected words to their word stem. Eg: Finally, final, finalized = fina
6. What are the cons of stemming? No meaning or context
7. Where it is being used? Sentiment analysis and spam classifier
8. What is lemmatization and provide an example?
9. Why stop words are being avoided? Repeated frequently and don’t play a major role while solving a problem
10. What is word embedding? Representation of words for test analysis in the form of real valued vectors. Encodes the meaning of the words such that words that are closer in the vector space are expected to be similar in meaning.
11. What are the types of word embedding? Words -> vector and sentence -> vector
12. Count or frequency: One hot encoding, BOW, TF-IDF
13. Deep Learning Trained Model: Word2Vec, Avg word2vec
14. What is one hot encoding?
15. Advantage and disadvantage of OHE?
    1. Pros - Easy to implement with python
    2. Cons – Sparse matrix -> overfitting, ML algorithm – fixed size inputs, no semantic meaning is getting captured, out of vocabulary.
16. What is Bag of words? Here all the words are given equal weightage. No major difference between words. Pros: Simple and intuitive, fixed size inputs -> ML algorithms. Cons: Sparse matrix or array -> Overfitting, ordering of the word is getting changed, out of vocabulary, semantic meaning is still not captured.
17. What are Ngram? Bigrams and trigrams
18. When to use BOW? If you have less data, go with bag of words. Useful for sentiment analysis to find which word is important. If you have huge dataset, then go for TF-IDF.
19. What is term frequency? No. of repetition of words in a sentence/ No. of words in the sentence
20. What is Inverse document frequency? Log(No. of sentences/ No of sentences containing words)
21. What is TF- IDF? Pros: Intuitive, fixed size -> vocab size, words important is getting captured. Cons: Sparsity still exists, Out of vocabulary
22. What are the problems with BOW and TF-IDF? Semantic information is not stored, gives importance to uncommon words, chances of overfitting. To overcome this, we have word2Vec.
23. What are the various techniques in Word2Vec? CBOW and Skipgram
24. What is Word2Vec? Uses NN to learn word associations from large corpus of text. Once trained, a model can detect synonymous words or suggest additional words for a partial sentence. Each distinct word with a particular list of numbers is called vector. Model where each word is represented as a vector of 32 or more dimension instead of single number. Semantic information and relation between different words is also preserved.
25. What are the steps to create words to vectors? Tokenization, create histograms, take most frequent words, create a matrix with all unique words and represent the occurrence relation between the words.
26. How it is represented into vector? Using vocabulary. There are unique words in corpus. We have so many features. Over each of the vocabulary, there will be a real value vectors provided against each feature.
27. What is cosine similarity?
28. What are transformers? Encoders and decoders
29. How does the transformer architecture overcome the challenges faced by traditional sequence to sequence models?

Overcomes challenges by several ways:



* Parallelization: Seq2Seq models process sequentially, slowing training. Transformers use self-attention to process token in parallel, speeding up both training and inference.
* Long range dependencies: Seq2Seq models struggle with this. Transformers capture these effectively with self-attention, allowing the model to focus on any part of the sequence, regardless of distance.
* Positional encoding: Since transformers process the entire sequence at once, positional encoding is used to ensure the model understands token order.
* Efficiency and Scalability: Seq2Seq Models are slower to scale due to sequential processing. Transformers, with their parallelism, scale better for large datasets and long sequences.
* Context Bottleneck: Se2Seq uses a single context vector, limiting information flow. Transformers let the decoder attend to all encoder outputting, improving context retention.

1. What is inside encoder? 2 layers – Self attention – Derived from attention is all you need
2. To which layer, embedding vectors are passed in transformers? Self-attention layer.
3. What is self-attention? Wonder how does computer understand “it”, “They” pronouns words. Whenever input text is pass, it has to select those important words. First, it creates three weights (randomly initialized). Three – queries, key, and values. These will be changed during back propagation. Second: Calculate score – multiply queries with keys. Third: Divide by 8 since we have dimension 64. Fourth: Apply softmax function – to understand the most important words based on probability. Fifth: Multiply softmax with value.
4. What is the problem of single head attention? Gives importance to only one word.
5. What is the use of multi head attention? Helps to gain attention/ importance of other words. How familiar it is with other words.
6. What is the use of positional encoding? Ordering of the words is important when you translate the sentence. This basically calculates the distance between the words.
7. Can you explain the concept of attention mechanisms in transformer models?
8. What is positional encoding in the context of LLMs models?   
   Address inability of transformers architectures to capture sequence order. Since transformers process tokens simultaneously through self-attention, they are unaware of token order. This positional encoding provides necessary information to help model understand the sequence of words.
9. What is Multi-head attention?   
   Enhancement of single head attention, allowing a model to attend to information from different representation subspaces simultaneously, focusing on various position in the data. Instead of using single attention mechanism, multi-head attention projects the queries, keys and values into multiple subspaces (denoted as h times) through distinct learned linear transformations.



This process involves applying the attention function in parallel to each of these projected versions of the queries, keys and values, which generates multiple output vectors. These outputs are then combined to produce the final dv-dimensional result. This approach improves the model’s ability to capture more complex patterns and relationships in the data.

1. What is RAG?
2. What is vector database?
3. How do you select which embedding model to use? Impact the overall success of the RAG system. Encoder selection to generate embedding is important or else it results in poor retrieval. Selection criteria: Vector dimension, performance evaluation – retrieval performance, model size, type – private or public, indexing cost, querying cost, storage cost, search latency, language support, privacy concerns, granularity of text.
4. What are the various vector databases? Chrom DB, Pinecone, open search, Cassandra
5. What is prompt engineering? Communicate with LLM to get desire output response. Text describes what the AI demands to do. Guided by this input, the AI generates an output. It is all about engineering your own AI prompt. Guide AI models like human interactions.
6. What is prompts? Detailed description of desired output expected from the model. Interactions between a user and the AI model. Quality is measured by how much detail you can provide.
7. Where is prompt engineering being used? Text summarization, question, and answering, code generations and information extraction.
8. Principles of prompt engineering? Write clear and specific instructions. Given the model time to think.
   1. Use delimiters to distinguish the data from the prompt
   2. Ask the structed output – json, xml and html etc
   3. Include style information to modify the tone of the output
   4. Give conditions to the model and ask if they are verified
   5. Give successful example of completing task then ask
   6. Specify the steps required to perform a task
   7. Instruct model to work out its own solution before giving answers.
   8. Iterate and refine your prompts
9. What are the various prompting techniques?
   1. Role
   2. Few shot
   3. Chain of thought
   4. Zero shot chain of thought
   5. Least to most
   6. Dual prompt
   7. Combing
10. What are the tips to design a good quality prompt?
    1. Role playing
    2. Clearness
    3. Specification
    4. Consistency
11. What are the elements of prompt?
    1. Instruction
    2. Context
    3. Input data
    4. Output indicator
12. What to avoid when creating prompts?
    1. Information overload
    2. Open ended questions
    3. Poor use of constraints
13. What do you evaluate RAG pipeline?
    1. RAGAS
    2. ARES
    3. Evaluation metrics – chunk attribution, chunk utilization, completeness, context adherence.
14. What are the challenges in RAG pipeline?

1. Missing Content: Highlighting strategies like data cleaning and better prompting to ensure informative responses.

2. Missed Top-Ranked Documents: Discusses the value of hyperparameter tuning and reranking to improve document retrieval accuracy.

3. Contextual Limitations: Suggests tweaking retrieval strategies and fine-tuning embeddings for richer context.

4. Extraction Challenges: Offers solutions such as prompt compression and data cleaning to enhance extraction accuracy.

5. Format Misalignment: Advises on better prompting, output parsing, and the use of Pydantic programs for format-specific outputs.

6. Incorrect Specificity: Talks about employing advanced retrieval strategies for precision in responses.

7. Incomplete Outputs: Recommends query transformations for comprehensive answers, especially for comparison questions.

8. Scalability Issues in Data Ingestion: Introduces parallelizing the ingestion pipeline as a solution for managing large data volumes.

9. Structured Data QA: Explores innovative packs like Chain-of-table and Mix-Self-Consistency for effective structured data question answering.

10. Complex PDF Data Extraction: Suggests specialized retrievers for extracting data from embedded tables in PDFs.

11. Fallback Models: Discusses implementing fallback models like Neutrino router for uninterrupted model access.

12. LLM Security: Touches on using tools like Llama Guard as a safety mechanism for preventing prompt injections and ensuring secure LLM interactions.

1. What are LLMs?
2. What is the main difference between LLMs and traditional statistical language models?
3. How do you measure the performance of an LLMs?
4. What do you know about LLMops?
5. Name some Open source LLMs:

* Llama 2
* Google Gemma

How these models are trained?

1. When to use open source or paid LLM? - It depends upon the data security, use cases.
2. Name some Paid LLMs:

* Open AI
* Cloude 3,
* Serverless Amazon Bedrock - provide APIs to access all these kinds of LLM models, cost is very less

1. Frameworks

* Langchain - Langserve, LangSmith
* Llama index

How can you work with it?

What are the functionalities?

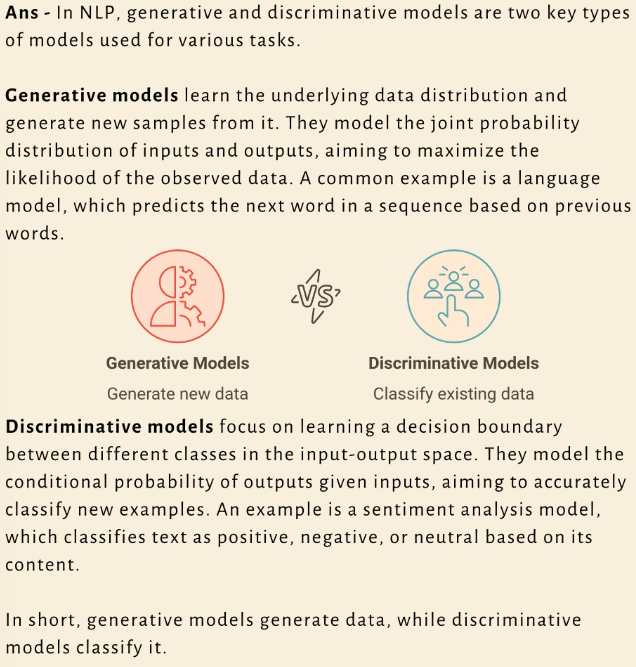
Difference between them?

1. You are working on an LLM, and it starts generating offensive or factually incorrect outputs. How would you diagnose and address this issue?
2. What are some common challenges associated with using LLMs?
3. How can catastrophic forgetting be mitigated in LLMs?

Catastrophic forgetting happens when an LLM forgets previously learned tasks while learning new ones, which limits its versatility. To mitigate this, several strategies are used:

* Rehearsal methods: Involve retraining model on a mix of old and new data, helping it retain knowledge of previous tasks.
* Elastic weight consolidation: This method assigns importance to certain model weights, protecting critical knowledge while learning new tasks.
* Modular approaches: Like progressive neural networks (ProgNet) and optimized fixed expansion layers introduces new modules for new tasks, allowing the LLM to learn without overwriting prior knowledge.

1. What are generative and discriminative models?



Overview

- 20% - Basic Concepts

- 30% - Machine Learning and Deep Learning

- 50% - NLP and Gen AI stuff