Mastering LLM

Top 40 LLM Interview Questions & Answers (Updated)

www.masteringllm.com

Which technique helps mitigate bias in prompt-based learning?

- Fine-tuning
- Data augmentation
- Prompt calibration
- Gradient clipping

Explanation

Prompt calibration involves adjusting prompts to minimize bias in the generated outputs. Fine-tuning modifies the model itself, while data augmentation expands the training data. Gradient clipping prevents exploding gradients during training.

www.masteringllm.com

LLM Interview Course

Want to Prepare yourself for an LLM Interview?

- ✓ 100+ Questions spanning 14 categories with Real Case Studies
- ✓ Curated 100+ assessments for each category
- ✓ Well-researched real-world interview questions based on FAANG & Fortune 500 companies
- ✓ Focus on Visual learning
- ✓ Certification

Coupon Code - LLM50

Coupon is valid till 30th Sep 2024

Do you need to have a vector store for all your text-based LLM use cases?

Yes

No

Explanation

A vector store is used to store the vector representation of a word or sentence. These vector representations capture the semantic meaning of the words or sentences and are used in various NLP tasks.
However, not all text-based LLM use cases require a vector store. Some tasks, such as summarization, sentiment analysis, and translation, do not need context augmentation.

Here is why:

- **Summarization**: This task involves condensing a larger body of text into a short summary. It does not require the context of other documents or sentences beyond the text being summarized.
- **Sentiment Analysis**: This task involves determining the sentiment (positive, negative, neutral) expressed in a piece of text. It is typically done based on the text itself without needing additional context.
- **Translation**: This task involves translating text from one language to another. The context is usually provided by the sentence itself and the broader document it is part of, rather than a separate vector store.

Which of the following is NOT a technique specifically used for aligning Large Language Models (LLMs) with human values and preferences?

- RLHF
- Direct Preference Optimization
- Data Augmentation

Explanation

Data Augmentation is a general machine learning technique that involves expanding the training data with variations or modifications of existing data. While it can indirectly impact LLM alignment by influencing the model's learning patterns, it's not specifically designed for human value alignment.

Incorrect Options:

A) **Reinforcement Learning from Human Feedback (RLHF)** is a technique where human feedback is used to refine the LLM's reward function, guiding it towards generating outputs that align with human preferences.

B) **Direct Preference Optimization (DPO)** is another technique that directly compares different LLM outputs based on human preferences to guide the learning process.

In Reinforcement Learning from Human Feedback (RLHF), what describes "reward hacking"?

Optimizes for desired behavior

Exploits reward function

Explanation

Reward hacking refers to a situation in RLHF where the agent discovers unintended loopholes or biases in the reward function to achieve high rewards without actually following the desired behavior. The agent essentially "games the system" to maximize its reward metric.

Why Option A is Incorrect:

While optimizing for the desired behavior is the intended outcome of RLHF, it doesn't represent reward hacking. Option A describes a successful training process. In reward hacking, the agent deviates from the desired behavior and finds an unintended way to maximize the reward.

Fine-tuning GenAI model for a task(e.g- Creative writing), which factor significantly impacts the models ability to adapt to the target task?

- Size of fine-tuning dataset
- Pre-trained model architecture

Explanation

The architecture of the pre-trained model acts as the foundation for fine-tuning. A complex and versatile architecture like those used in large models (e.g., GPT-3) allows for greater adaptation to diverse tasks. The size of the fine-tuning dataset plays a role, but it's secondary. A well-architected pre-trained model can learn from a relatively small dataset and generalize effectively to the target task.

Why A is Incorrect:

While the size of the fine-tuning dataset can enhance performance, it's not the most crucial factor. Even a massive dataset cannot compensate for limitations in the pre-trained model's architecture. A well-designed pre-trained model can extract relevant patterns from a smaller dataset and outperform a less sophisticated model with a larger dataset.

What does the self-attention mechanism in transformer architecture allow the model to do?

- Weigh word importance
- Predict next word
- Automatic summarization

Explanation

The self-attention mechanism in transformers acts as a spotlight, illuminating the relative importance of words within a sentence.

In essence, self-attention allows transformers to dynamically adjust the focus based on the current word being processed. Words with higher similarity scores contribute more significantly, leading to a richer understanding of word importance and sentence structure. This empowers transformers for various NLP tasks that heavily rely on context-aware analysis.

Incorrect Options:

Predict next word: While transformers can be used for language modeling (including next-word prediction), this isn't the primary function of self-attention.

Automatic summarization: While self-attention is a core component of summarization models, it's not solely responsible for generating summaries.

What is one advantage of using subword algorithms like BPE or WordPiece in Large Language Models (LLMs)?

- Limit vocabulary size
- Reduce amount of training data
- Make computationally efficient

Explanation

LLMs deal with massive amounts of text, leading to a very large vocabulary if you consider every single word. Subword algorithms like Byte Pair Encoding (BPE) and WordPiece break down words into smaller meaningful units (subwords) which are then used as the vocabulary. This significantly reduces the vocabulary size while still capturing the meaning of most words, making the model more efficient to train and use.

Incorrect Answer Explanations:

Reduce amount of training data: Subword algorithms don't directly reduce the amount of training data. The data size remains the same.

Make computationally efficient: While limiting vocabulary size can improve computational efficiency, it's not the primary purpose of subword algorithms. Their main advantage lies in effectively representing a large vocabulary with a smaller set of units.

Compared to Softmax, how does Adaptive Softmax speed up large language models?

- Sparse word reps
- Zipf's law exploit
- Pre-trained embedding

Explanation

Standard Softmax struggles with vast vocabularies, requiring expensive calculations for every word. Imagine a large language model predicting the next word in a sentence. Softmax multiplies massive matrices for each word in the vocabulary, leading to billions of operations! Adaptive Softmax leverages Zipf's law (common words are frequent, rare words are infrequent) to group words by frequency. Frequent words get precise calculations in smaller groups, while rare words are grouped together for more efficient computations. This significantly reduces the cost of training large language models.

Incorrect Answer Explanations:

(A) Sparse word reps: While sparse representations can improve memory usage, they don't directly address the computational bottleneck of Softmax in large vocabularies.

(C) Pre-trained embedding: Pre-trained embeddings enhance model performance but don't address the core issue of Softmax's computational complexity.

Which configuration parameter for inference can be adjusted to either increase or decrease randomness within the model output layer?

- Max new tokens
- Top-k sampling
- Temperature

Explanation

During text generation, large language models (LLMs) rely on a softmax layer to assign probabilities to potential next words. Temperature acts as a key parameter influencing the randomness of these probability distributions.

Lower Temperature: When set low, the softmax layer assigns significantly higher probabilities to the single word with the highest likelihood based on the current context.
Higher Temperature: A higher temperature "softens" the probability distribution, making other, less likely words more competitive.

Why other options are incorrect:
(A) **Max new tokens**: This parameter simply defines the maximum number of words the LLM can generate in a single sequence.
(B) **Top-k sampling**: This technique restricts the softmax layer to consider only the top k most probable words for the next prediction.

What transformer model uses masking & bi-directional context for masked token prediction?

- Autoencoder
- Autoregressive
- Sequence-to-sequence

Explanation

Autoencoder models are pre-trained using masked language modeling. They use randomly masked tokens in the input sequence and the pretraining objective is to predict the masked tokens to reconstruct the original sentence.

What technique allows you to scale model training across GPUs when the model doesn't fit in the memory of a single chip?

DDP

FSDP

Explanation

FSDP (Fully Sharded Data Parallel) is the technique that allows scaling model training across GPUs when the model is too big to fit in the memory of a single chip. FSDP distributes or shards the model parameters, gradients, and optimizer states across GPUs, enabling efficient training.

Incorrect Answers:

A) DDP (Distributed Data-Parallel) is a technique that distributes data and processes batches in parallel across multiple GPUs, but it requires the model to fit onto a single GPU.

What is the purpose of quantization in training large language models?

- Reduce memory usage
- Improve model accuracy
- Enhance model interpretability

Explanation

Quantization helps reduce the memory required to store model weights by reducing their precision.

Incorrect options:

b) Improve model accuracy: While quantization can have some impact on model accuracy, its primary purpose is to reduce memory usage.

C) Enhance model interpretability: Quantization does not directly enhance model interpretability.

How can scaling laws be used to design compute optimal models? By

- Optimizing model & data size
- Improve model interpretability
- Reduce training time
- Enhance model scalability

Explanation

Scaling laws provide valuable insights into the relationship between model size (number of parameters), dataset size, and the model's performance (often measured as loss). This relationship can be expressed mathematically through power laws.

Here's how scaling laws help design compute-optimal models:

- **Understanding cost trade-offs:** By analyzing scaling laws, you can estimate the impact of increasing model size or dataset size on performance and computational resources (training time, memory usage). This allows you to find a balance between model complexity and training cost.
- **Targeted optimization:** You can use scaling laws to predict the performance gain from increasing model size or data size. This helps you focus optimization efforts on the factors that will have the most significant impact on performance within your computational budget.

What's catastrophic forgetting in fine-tuning?

- Other tasks perform worse
- All tasks perform better
- Pre-trained weights enhance

Explanation

Catastrophic forgetting refers to the degradation of performance on tasks other than the one being fine-tuned, as the weights of the original model are modified.

Incorrect options:

b) All tasks perform better: This is incorrect as catastrophic forgetting leads to a loss of performance on other tasks.
c) Pre-trained weights enhance: This is incorrect as catastrophic forgetting occurs due to the modification of weights during fine-tuning.

Parameter Efficient Fine-Tuning (PEFT) updates only a small subset of parameters and this helps prevent catastrophic forgetting

True

False

Explanation

Parameter Efficient Fine-Tuning (PEFT) is a method that updates only a small subset of parameters during the fine-tuning process. This approach is designed to be more memory efficient and to prevent catastrophic forgetting. Catastrophic forgetting is a phenomenon where a neural network forgets its previously learned information upon learning new information. By updating only a small subset of parameters, PEFT mitigates this issue, allowing the model to retain its previously learned knowledge while adapting to new tasks.

Explanation for the incorrect answer (False): If you chose False, the misunderstanding might be due to the assumption that all parameters need to be updated during fine-tuning. However, in PEFT, only a small subset of parameters is updated. This is indeed an effective strategy to prevent catastrophic forgetting and is not less efficient. It allows the model to maintain its general knowledge while adapting to specific tasks, thereby enhancing its performance on those tasks without a significant increase in computational cost.

You can use an algorithm other than Proximal Policy Optimization to update the model weights during RLHF

- True
- False

Explanation

For instance, you can use an algorithm called Q-Learning. PPO is the most popular for RLHF because it balances complexity and performance, but RLHF is an ongoing field of research and this preference may change in the future as new techniques are developed.

In a Transformer model with group attention, how does the mechanism differ from standard self-attention when processing a sentence?

- Replaces self-attention
- Pre-defined word groups
- Attention on specific word

Explanation

Standard self-attention in a Transformer considers the relationships between individual words within a sentence. Group attention, on the other hand, introduces a new layer of attention. This layer focuses on groups of words pre-defined based on specific criteria, such as syntactic or semantic groupings (e.g., noun phrases, verb phrases).

During LLM training, which step is NOT directly involved in the process?

- Feature engineering
- Pre-training
- Fine-tuning
- RLHF

Explanation

LLMs primarily rely on the raw text data itself for training. Feature engineering, which involves manually extracting specific features from the data, is not a typical step in LLM training. Options (b), (c), and (d) are all common stages in the LLM training pipeline.

Pre-training is a crucial step in LLM training. What is the main objective of pre-training?

- To perform a specific task
- General language understanding

Explanation

Pre-training aims to equip the LLM with a foundational understanding of language by exposing it to a vast amount of text data. This allows the model to learn general representations of words, their relationships, and overall language structure

Which of the following sequences represents the most likely order of LLM Training stages?

A. Pre-training

B. RLHF

C. Instruction Fine-tuning

A -> C -> B

B -> A -> C

C -> A -> B

Explanation

LLM training follows a specific order:

Pre-training (A): The LLM is exposed to a massive dataset to learn general language patterns and relationships between words.

Instruction Fine-tuning (C): The pre-trained model is adapted to a specific task using labeled data and instructions. This tailors the model's knowledge to the desired task.

RLHF (Reinforcement Learning from Human Feedback) (B): This optional stage further refines the model's behavior by incorporating human feedback through a reward system. The LLM receives rewards for desirable outputs.

A technique that utilizes a smaller model to learn from a larger pre-trained model, improving efficiency, is called:

- Gradient Clipping
- Backpropagation
- Knowledge Distillation
- Batch Normalization

Explanation

Knowledge distillation is a technique that allows a smaller model (student) to learn from a larger, pre-trained model (teacher). It improves training efficiency by leveraging the knowledge already encoded in the teacher model.

Here's how knowledge distillation works:

1. Train a large, powerful model (teacher) on a massive dataset.
2. During training of the smaller model (student), instead of relying solely on the original loss function (difference between predicted and actual labels), the student also learns from the teacher's outputs or internal representations.
3. This "distilled knowledge" guides the student model towards learning similar patterns and achieving good performance, but with less data and computational resources compared to training from scratch.

Which method places ## at the start of tokens?

- BPE
- WordPiece

Explanation

WordPiece tokenization method places ## at the beginning of tokens. This is a characteristic feature of WordPiece.

Which technique uses gating functions to decide which model to use based on the input?

- Ensemble Techniques
- Mixture of Experts (MoE)

Explanation

Mixture of Experts (MoE) is a machine learning technique that uses multiple models, called “experts”, and a gating function that decides which expert to use based on the input. This allows MoE to model more complex patterns and adapt to different regions of the input space, making it more flexible than traditional ensemble techniques, which typically combine predictions from multiple models without such a gating function. Therefore, option B is correct. The other option is incorrect because ensemble techniques do not use gating functions. They use multiple models and combine their predictions.

What does ‘Prompt leaking’ signify in the context of Language Learning Models (LLMs)?

Extracting sensitive Info

Hijacking model's output

Explanation

‘Prompt leaking’ in the context of Language Learning Models (LLMs) refers to the act of extracting sensitive or confidential information from the model’s response. This could potentially be exploited by adversaries to gain unauthorized insights into the LLM’s behavior or compromise its security.

For example, consider a scenario where an LLM is trained on a dataset that includes confidential company emails. If a user asks the model to generate a response based on a specific topic covered in those emails, the model might inadvertently include sensitive information in its response. This is an instance of ‘Prompt leaking’, as the model has leaked information that was supposed to remain confidential

Which database would you use if you want to store Multi-dimensional vectors and perform ANN search?

- Vector Database
- Traditional Database
- Explanation

Traditional databases, like relational databases, are designed to store data in tables with rows and columns. This structure is not efficient for storing and searching multi-dimensional vectors.

On the other hand, vector databases specialize in handling high-dimensional vectors. They are optimized for:

Storage: Vectors are efficiently stored and compressed within the database.
ANN Search (Approximate Nearest Neighbor Search): This allows you to find data points similar to a query vector. Vector databases use specialized algorithms to perform these searches efficiently, even in high-dimensional spaces.

In summary, for storing multi-dimensional vectors and performing ANN search, a vector database is the most suitable choice due to its optimized storage and search capabilities for this specific type of data.

Which of the following vector indexing techniques relies on grouping similar vectors in a cluster for efficient retrieval?

- Flat Indexing
- Inverted File Index
- Principal Component Analysis

Explanation

(a) Flat Indexing: This stores vectors without any specific organization. While it can be used for similarity searches, it's not efficient for large datasets.

(b) Inverted File Index: This technique is commonly used for text retrieval in document databases. It indexes words and keeps track of which documents contain those words. While it can be adapted for vector similarity search with specific techniques, it doesn't inherently group similar vectors in clusters.

(c) Principal Component Analysis (PCA): This technique reduces the dimensionality of vectors while preserving the most important information. While it can be used for dimensionality reduction before indexing, PCA doesn't involve clustering similar vectors.

For a small review dataset, if you want a 100% recall rate which vector index you would use? Speed is not consideration here.

- Flat Index
- HNSW
- Random Projection

Explanation

In a small dataset, a flat index allows for an exhaustive search, comparing each review vector to every other vector. This maximizes the chance of finding the most similar reviews with perfect accuracy.

Incorrect Option:

B. HNSW:

While HNSW can be accurate, it might not guarantee finding the absolute closest neighbors in every case due to the focus on efficient search within clusters.

C. Random Projection:

Similar to HNSW, the potential loss of information due to dimensionality reduction might compromise the goal of perfect accuracy

In Inverted File Index (IVF) index which parameter you would tune to expand number of clusters?

nprob

nlist

Explanation

nlist: This parameter controls the number of vectors assigned to each inverted list during the initial clustering stage. Increasing nlist leads to the creation of more inverted lists, which essentially represent more clusters.

Incorrect Answer:
This parameter determines the number of probes (comparisons) performed within each inverted list during retrieval. It doesn't directly affect the number of clusters but rather influences how many elements within each cluster are explored during search.

Which metric is NOT typically used for evaluating the quality of factual language summaries generated by an LLM?

- ROUGE Score
- BLEU Score
- Perplexity

Explanation

Perplexity is a measure of how well the model predicts the next word in a sequence. While it might be a relevant metric for some LLM tasks, it's not commonly used for evaluating factual language summaries. Options (a), (b) are all common metrics for assessing the quality and factual accuracy of summaries generated by LLMs.

Which of the following indices represents a method that involves multiplying with another metric to reduce the size of the original vector?

- Random Projection Index
- Flat Index

Explanation

The Random Projection Index is a technique used in dimensionality reduction. It works by projecting the original high-dimensional data into a lower-dimensional space using a random matrix. This process involves multiplication with another metric (the random matrix), effectively reducing the size of the original vector. On the other hand, a Flat Index does not involve such a process.

What's the right pre-filtering order in a vector database?

Meta-data filtering --> Top-K

Top-K --> meta-data filtering

Explanation

In the context of a vector database, pre-filtering typically involves two steps: performing meta-data filtering and then retrieving the top K results from the vector index. The correct sequence is to first perform meta-data filtering, which reduces the overall search space, and then execute the vector query on these filtered vectors to generate the top-k results.

What's the right post-filtering order in a vector database?

Meta-data filtering --> Top-K

Top-K --> meta-data filtering

Explanation

In the context of a vector database, post-filtering typically involves two steps: retrieving the top K results from the vector index and then performing meta-data filtering. The correct sequence is to first perform tok-k results from vector index to reduce search space, and then execute meta-data filtering to general final top-k results.

Which type of attention provides the best overall accuracy & speed?

- Single-headed Attention
- Multi-query Attention
- Grouped Query Attention

Explanation

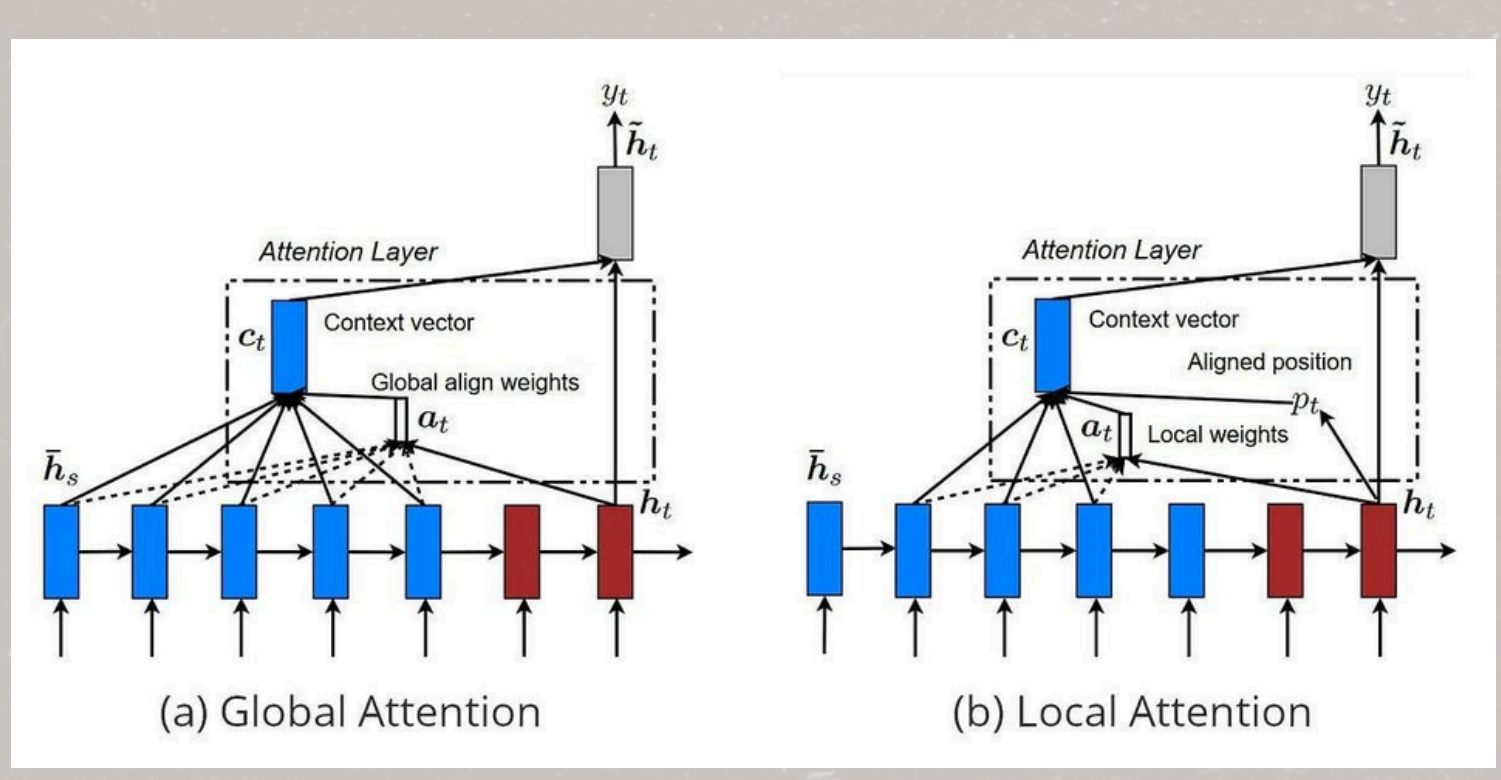
GQA allows the attention heads to pay attention to different parts of the sequence, while still allowing for better speed and compute efficiencies compared to multi-headed attention.

What is the key difference between Global and Local Attention mechanisms in LLMs?

- Global uses all hidden states
- Local uses all hidden states
- Global focuses on subsets

Explanation

Global attention involves utilizing all hidden states to generate the context vector, which can be more computationally intensive. In contrast, local attention only considers a subset of hidden states, making it more efficient.



What is the purpose of Positional Encoding in a Transformer model?

- To normalize the data
- Introduce sequence order
- To reduce overfitting

Explanation

Since Transformers do not inherently consider token order, Positional Encoding adds information about the position of tokens in the sequence.

Which process is used to convert text into high-dimensional numerical representations in LLMs?

- Tokenization and Embeddings
- Backpropagation
- Data Augmentation

Explanation

LLMs use tokenization to break text into tokens, which are then converted into embeddings that capture semantic meaning.

Which prompt engineering technique is used to ask for step-by-step reasoning?

- Zero-shot prompting
- Chain of thought prompting
- Few-shot prompting

Explanation

Chain of thought prompting (B) is a technique where the prompt is crafted to encourage a model to generate a sequence of intermediate steps or reasoning that leads to the final answer. This is particularly useful in complex problem-solving tasks where the direct generation of an answer might be difficult without breaking down the problem into more manageable parts. By explicitly asking the model to describe its reasoning step by step, the prompt helps simulate a thought process that can improve the model's accuracy and the transparency of its conclusions.

What type of questions does GraphRAG excel at answering compared to baseline RAG?

- Factual questions
- Aggregation of information

Explanation

GraphRAG excels at answering questions that require aggregation of information across the dataset, as it organizes data into meaningful semantic clusters, enabling it to summarize themes effectively.

How does AgenticRAG improve on basic RAG?

- Use of LLMs
- Autonomous info seeking
- Faster processing

Explanation

AgenticRAG improves on RAG by enabling autonomous information seeking. It can decide when to retrieve more info, ask follow-up questions, or clarify ambiguities. This makes it more dynamic and adaptive than traditional RAG.

Incorrect Explanations:

A) Larger models are not the main improvement.

C) Faster processing is important but not the key difference.

How does a planner agent enhance the capabilities of AgenticRAG?

- Responding in real-time
- Breaking down complex queries

Explanation

A planner agent enhances the capabilities of AgenticRAG by breaking down complex queries into manageable sub-queries. This allows the planner agent to execute these sub-queries across various RAG pipelines linked to different data sources, ultimately merging the results to form a coherent final response. This approach facilitates more efficient processing of intricate queries.

What is the primary function of a routing agent in AgenticRAG?

- Summarize multiple documents
- Select downstream RAG pipeline

Explanation

The primary function of a routing agent in AgenticRAG is to select the appropriate downstream RAG pipeline based on the input query. It uses an LLM to analyze the query and determine which RAG pipeline (such as summarization or question-answering) is best suited to handle the request. This process is known as agentic reasoning and is essential for optimizing the use of available resources.

What distinguishes AgenticRAG from traditional RAG models?

- Uses dynamic retrieval
- RL during generation
- generate text w/o retrieval
- Explanation

AgenticRAG distinguishes itself from traditional RAG models by utilizing user feedback to dynamically adjust the retrieval process. This allows the model to better align with user preferences and improve the relevance of the retrieved information, enhancing the overall quality of the generated output.

What are the key steps involved in the Retrieval Augmented Generation (RAG) pipeline?

- Retrieval, Generation, Ranking
- Ranking,Generation,Retrieval
- Retrieval, Ranking, Generation

Explanation

The RAG pipeline involves three key steps:

1. **Retrieval**: The query is encoded and compared to pre-computed embeddings of text chunks from a corpus, retrieving relevant documents.
2. **Ranking**: Retrieved documents are ranked based on relevance to the query, refining the set for the final response.
3. **Generation**: The language model generates a response based on the query and the most relevant documents.

What is the term for the process of reducing the number of parameters in a model to improve its computational efficiency?

- Model Pruning
- Knowledge Distillation
- Model Compression

Explanation

Model pruning is the process of reducing the number of parameters in a model to improve its computational efficiency. This is done by identifying and removing redundant or less important parameters, which can significantly reduce the model's computational complexity without significantly affecting its performance.

What is order of execution when you set both top-k and top-p together.

- Top K acts before Top P
- Top P acts before Top K
- Top K does not matter
- Top P does not matter

Explanation

For each token selection step, the top-K tokens with the highest probabilities are sampled. Then tokens are further filtered based on top-P with the final token selected using temperature sampling.

True or False: The more fine-tuning data we can provide to fine-tune a LLM, the better the model performance.

True

False

Explanation

The most important factor to consider when providing fine-tuning data examples is their quality and the diversity of downstream queries users might submit to the LLM. If the downstream use cases are diverse, then we need more fine-tuning data. While providing more fine-tuning data might typically boost the model performance, it's not always true. The AlpaGasus paper has shown that highly curated high-quality datasets are much more important than having large datasets to produce a better model.

In LLM decoding, which strategy considers several possible continuations of a sequence at the same time?

- Greedy decoding
- Beam search

Explanation

Beam search is a decoding strategy in LLMs that addresses a limitation of greedy decoding. It works by:

1. Starting with the first word in the prompt.
2. Predicting the top 'k' most probable continuations (beams) for the next word.
3. Evaluating each beam based on a combination of probabilities in the sequence so far.
4. Selecting the top 'k' beams to continue exploring and predicting the next word for each remaining beam.
5. Repeating steps 2-4 until the desired length is reached.

www.masteringllm.com

LLM Interview Course

Want to Prepare yourself for an LLM Interview?

- ✓ 100+ Questions spanning 14 categories with Real Case Studies
- ✓ Curated 100+ assessments for each category
- ✓ Well-researched real-world interview questions based on FAANG & Fortune 500 companies
- ✓ Focus on Visual learning
- ✓ Certification

Coupon Code - LLM50

Coupon is valid till 30th Sep 2024

www.masteringllm.com

AgenticRAG with LlamaIndex 

Want to learn why AgenticRAG is future of RAG?

- Master **RAG fundamentals** through practical case studies
- Understand how to overcome **limitations of RA**G
- Introduction to **AgenticRAG** & techniques like **Routing Agents, Query planning agents, Structure planning agents, and React agents with human in loop**.
- **5** real-time **case studies with code walkthroughs**