Theoretical Questions

Generative AI:

1. Explain how Diffusion Models work in Generative AI.

In generative AI, diffusion models gradually convert noise into structured data (such as text or images) by using a stepwise denoising procedure. There are two stages to the main idea:

i. Forward Process (Diffusion)
   - begins with an actual image (or data) and gradually introduces Gaussian noise over a number of stages.
   - The image turns into almost pure noise after a sufficient number of steps.
   - Since this process is represented as a Markov chain, corruption is ensured to occur gradually and smoothly.

ii. Reverse Process (Denoising)
   - The model gradually anticipates and eliminates noise, learning to reverse the diffusion process.
   - use a neural network that has been trained to predict noise at every stage, such as U-Net.
   - The model creates a fresh image from pure noise by iterative denoising.

Why it works well ?

- High-quality generation: Captures complex distributions.
- Flexibility: Can generate diverse outputs.
- Stability: More stable than GANs, avoiding mode collapse.

Popular Models :-

- DALL·E 2 (Text-to-Image)
- Stable Diffusion
- Imagen by Google

2. What are LoRA adapters, and how do they improve fine-tuning LLMs?

For large language models (LLMs), LoRA (Low-Rank Adaptation) adapters are a parameter-efficient fine-tuning technique. LoRA maintains the original model weights frozen while inserting tiny, trainable low-rank matrices into specific layers (usually the attention layers) in place of updating all model parameters. This method maintains the efficiency of large models while drastically lowering the computational and memory burden associated with fine-tuning them.

How LoRA Improves Fine-Tuning of LLMs :-

A. Reduced Memory and Computational Cost:
   - All model parameters are updated during traditional fine-tuning, which uses a lot of memory and GPU power.
   - LoRA makes fine-tuning possible even on hardware with low resources because it only modifies a few other parameters.

B. Maintaining Pre-Trained Knowledge:
   - LoRA makes sure that the LLM's fundamental information is preserved because the basic model stays fixed.
   - The adapters acquire task-specific changes without changing the general knowledge that underlies them.

C. Faster and More Efficient Training:
   - The low-rank matrices in LoRA significantly reduce the number of parameters to optimize, leading to faster convergence.
   - This enables effective domain adaptability without requiring a lot of retraining.

D. Multi-Task Adaptation and Modularity:
   - The same base model can be used to train several LoRA adapters independently for various tasks.
   - This allows for flexible deployment across a range of applications by eliminating the need to store multiple copies of big LLMs.

E. Improved Scalability for Real-World Use Cases:
   - LoRA makes it possible to modify LLMs for use in sectors including healthcare, finance, and law.
   - It’s modular design makes it perfect for implementing AI helpers that must quickly adjust to new data.

3. Compare GANs and VAEs for image generation.

GANs (Generative Adversarial Networks) :-

- GANs employ two networks: a discriminator to distinguish between authentic and fraudulent images, and a generator to produce images.
- They produce crisp, high-quality photos, but their mode collapse frequently restricts variation.
- Adversarial loss makes training unstable, necessitating meticulous hyperparameter adjustment.
- GANs do not explicitly learn a structured latent space, which makes controlled generation difficult.
- They are frequently employed for style transfer, deepfakes, and realistic picture creation.

VAEs (Variational Autoencoders) :-

- VAEs are made up of a decoder that reconstructs images and an encoder that compresses data into a latent space.
- Although they offer a variety of outputs, the reconstruction loss frequently results in blurrier images.
- Since training uses probabilistic loss instead of adversarial optimization, it is stable.
- VAEs learn a smooth latent space, enabling easy interpolation and controlled generation.
- They are commonly used for anomaly detection, data compression, and structured image generation.

Computer Vision:

1. **Explain the differences between CNN and Vision Transformers (ViTs).**

CNNs (Convolutional Neural Networks):

- CNNs create hierarchical feature maps from low-level to high-level structures by using convolutional filters to identify local patterns like edges and textures.
- They are effective at vision tasks because they have innate spatial inductive biases, which allow them to understand the structure and location of items in an image.
- CNNs are more computationally efficient than ViTs because they employ local connections and shared weights, which reduce the number of parameters.
- Even with smaller datasets, CNNs perform well because of their structured methodology, which makes them appropriate for real-world applications with sparse data.
- Without deeper architectures, it is challenging to capture relationships between distant areas of an image because CNNs primarily concentrate on local feature extraction.

ViTs (Vision Transformers):

- ViTs use self-attention techniques to concurrently capture local and global dependencies, treating images as a series of patches.
- Since ViTs lack the spatial hierarchies that CNNs possess, they can only learn spatial relationships from training data, which means they need more data to learn well.
- They are more computationally costly because they use self-attention to process the full image globally, which grows quadratically with image size.
- Large datasets and intensive pretraining (such as on ImageNet) are necessary for ViTs to function well because they lack CNNs' efficiency advantages while learning spatial features.

- When given enough data and training, ViTs outperform CNNs in terms of flexibility to a variety of datasets and exhibit higher robustness and generalization, particularly in large-scale vision tasks.

2. How does YOLOv11 compare to traditional object detection models?

You Only Look Once version 11, or YOLOv11, offers a number of improvements over past YOLO versions and more conventional object identification models, such as Faster R-CNN and SSD. It is a better option for real-time applications since it incorporates contemporary deep learning techniques to increase accuracy, speed, and adaptability.

YOLOv11 represents a significant advancement in real-time object detection, introducing several key innovations that distinguish it from traditional object detection models:

Architecture & Feature Extraction: While typical models rely on CNN-based architectures, which may have trouble recognizing small or overlapping objects, YOLOv11 employs a transformer-based backbone for improved global feature extraction.

Speed and Real-Time Performance: While more conventional models like Faster R-CNN and SSD require numerous processing stages, which slows them down, YOLOv11 is built for real-time detection with faster inference.

Training and Computational Efficiency: YOLOv11 eliminates the need for Non-Maximum Suppression (NMS), reducing computational overhead, while traditional models depend on NMS, which increases post-processing time.

Detection Accuracy and Robustness: YOLOv11 introduces dual-label assignment to handle overlapping objects more accurately, whereas traditional models often struggle with object detection in crowded scenes.

Adaptability to Different Environments: YOLOv11 is highly efficient for real-time applications like autonomous driving and surveillance, whereas traditional models require careful tuning and higher computational resources, making them less suitable for edge devices.

3. What are the challenges of OCR in low-light or complex backgrounds?

OCR (Optical Character Recognition) in low-light or complex backgrounds faces several challenges that impact accuracy and reliability.

One significant problem is **noise and poor contrast**, which causes text to merge into the backdrop because of glare or inadequate illumination, making it challenging for OCR models to discern between characters. Text readability is further diminished by the increased noise and blurriness that low-light circumstances frequently cause in photographs. Furthermore, OCR has trouble with backdrop clutter and distortions because intricate patterns, textures, or shadows can obstruct text detection, resulting in lost characters or false positives.

Another challenge is **variable fonts, sizes, and orientations**, especially in handwritten or stylized text, where characters may be irregularly shaped or rotated. Conventional OCR methods are less successful in real-world situations because they frequently make fixed assumptions about text alignment. Variability in language and script can also make things more complicated, especially when working with unique symbols, cursive writing, or multilingual material. Advanced preprocessing methods including adaptive thresholding, picture enhancement, and deep learning-based OCR models trained under various settings are sometimes needed to overcome these difficulties.