Generative AI & LLMs - Interactive Interview Cheat Sheet

1. Foundational Concepts

- Generative AI: AI that produces content such as text, images, audio, or code.

- LLMs (Large Language Models): Deep learning models trained on large text corpora to understand and generate human language.

- Transformer: Backbone architecture using attention mechanisms (e.g., BERT, GPT).

- Tokens: LLMs process inputs as sub-word units called tokens.

- Context Window: Max number of tokens model can process (e.g., GPT-4 ~8k to 128k tokens).

2. Prompt Engineering

- Zero-shot: Prompting the model without examples.

- One-shot: Providing one example with the prompt.

- Few-shot: Providing multiple examples to guide the model.

- Use parameters like temperature (controls randomness), max tokens (response length).

- Example: 'Translate the following English sentence to French: How are you?' -> 'Comment ça va ?'

3. Advanced Techniques

- RAG (Retrieval-Augmented Generation): Combines LLMs with external knowledge retrievers.

- PEFT (e.g., LoRA): Fine-tuning small number of parameters for efficiency.

- RLHF: Uses human feedback to align model outputs with user preferences.

- Tool Use: Models can invoke tools or APIs using structured output (e.g., OpenAI function calling).

4. Fine-tuning & Instruct-tuning

- SFT: Supervised Fine-Tuning using labeled instructions.

- Instruct-tuning: Training models to follow natural instructions.

- PEFT Methods: LoRA, Prefix Tuning, Adapter Tuning.

- Datasets: Alpaca, OASST, Dolly used for alignment and instruct training.

5. Evaluation & Debugging

- BLEU / ROUGE: Used for text generation comparisons.

Generative AI & LLMs - Interactive Interview Cheat Sheet

- Perplexity: Measures prediction uncertainty.

- F1 / Accuracy: Used in classification/NER tasks.

- Hallucination Tracing: Use LangSmith or custom trace logs to detect factual errors.

6. Deployment Tips

- Use FastAPI or Flask to serve your models as REST APIs.

- Optimize with ONNX / TensorRT for GPU efficiency.

- For large prompts, use token chunking or streaming.

- Use Docker for containerization and Kubernetes for orchestration on scale.

7. Key Libraries & Tools

- Hugging Face Transformers, Datasets, Accelerate

- LangChain, LlamaIndex, Haystack for orchestration & RAG

- Gradio / Streamlit: Quick UI for demos

- OpenAI / Cohere / Mistral APIs: Hosted model access

- Weights & Biases / MLflow: Experiment tracking and model monitoring