Research Summary

1. What is Large Language Models (LLMs)

A large language model (LLM) is a type of artificial intelligence (AI) program that can recognize and generate text, among other tasks. LLMs are trained on huge sets of data — hence the name "large." LLMs are built on machine learning: specifically, a type of neural network called a transformer model.

How does it work?

LLMs use a specific kind of machine learning called the transformer model.

In simpler words:
 An LLM is like a computer program that has seen so many examples of human language that it can now interpret and generate text.

What is Deep Learning?

LLMs rely on deep learning, a machine learning technique that analyzes data (like text) to find patterns.
 Deep learning helps the model understand how letters, words, and sentences work together.
 Eventually, it can recognize differences and relationships in content without any human help.

Importance:

- LLMs are based on machine learning and deep learning
- They analyze huge amounts of data to understand human language
- They use probability to identify patterns

Some human fine-tuning is done to ensure the final output is reliable.

2. What is Generative AI?

Generative AI enables users to quickly generate new content based on a variety of inputs. Inputs and outputs to these models can include text, images, sounds, animation, 3D models, or other types of data.

How Does Generative AI Work?

Generative AI models use neural networks to identify the patterns and structures within existing data to generate new and original content.

How to Evaluate Generative AI Models?

The three keys requirements of a successful generative AI model are:

1. **Quality:** Especially for applications that interact directly with users, having high-quality generation outputs is key.
   For example, in speech generation, poor speech quality is difficult to understand. Similarly, in image generation, the desired outputs should be visually indistinguishable from natural images.

2. **Diversity:** A good generative model captures the minority modes in its data distribution without sacrificing generation quality. This helps reduce undesired biases in the learned models.

3. **Speed:** Many interactive applications require fast generation, such as real-time image editing to allow use in content creation workflows.

What are the Benefits of Generative AI?

Generative AI is important for a number of reasons. Some of the key benefits of generative AI include:

1. Generative AI algorithms can be used to create new, original content, such as images, videos, and text, that's indistinguishable from content created by humans. This can be useful for applications such as entertainment, advertising, and creative arts.

2. Generative AI algorithms can be used to improve the efficiency and accuracy of existing AI systems, such as natural language processing and computer vision. For example, generative AI algorithms can be used to create synthetic data that can be used to train and evaluate other AI algorithms.

3. Generative AI algorithms can be used to explore and analyze complex data in new ways, allowing businesses and researchers to uncover hidden patterns and trends that may not be apparent from the raw data alone.

4. Generative AI algorithms can help automate and accelerate a variety of tasks and processes, saving time and resources for businesses and organizations.

Overall, generative AI has the potential to significantly impact a wide range of industries and applications and is an important area of AI research and development.

3. What is Agentic AI

Agentic AI is an artificial intelligence system that can accomplish a specific goal with limited supervision. It consists of AI agents—machine learning models that mimic human decision-making to solve problems in real time.

How Does Agentic AI Work?

Agentic AI uses a four-step process for problem-solving:

1. **Perceive**: AI agents gather and process data from various sources, such as sensors, databases and digital interfaces. This involves extracting meaningful features, recognizing objects or identifying relevant entities in the environment.
2. **Reason**: A large language model acts as the orchestrator, or reasoning engine, that understands tasks, generates solutions and coordinates specialized models for specific functions like content creation, visual processing or recommendation systems. This step uses techniques like retrieval-augmented generation (RAG) to access proprietary data sources and deliver accurate, relevant outputs.
3. **Act**: By integrating with external tools and software via application programming interfaces, agentic AI can quickly execute tasks based on the plans it has formulated. Guardrails can be built into AI agents to help ensure they execute tasks correctly. For example, a customer service AI agent may be able to process claims up to a certain amount, while claims above the amount would have to be approved by a human.
4. **Learn**: Agentic AI continuously improves through a feedback loop, or "data flywheel," where the data generated from its interactions is fed into the system to enhance models. This ability to adapt and become more effective over time offers businesses a powerful tool for driving better decision-making and operational efficiency.

4. Difference between Generative AI and Agentic AI

- **Generative AI** just creates new content (text, images, music, etc.).

- **Agentic AI** plans the task, collects data, and executes the work—making it more powerful and flexible than Generative AI.

Key Differences Between Agentic AI and Generative AI

Aspect	Generative AI	Agentic AI
Primary Purpose	The main goal is to generate content that resembles human-made creations, often used for artistic or communicative purposes.	Its purpose is to execute tasks, make decisions, and achieve specific objectives, without requiring constant human input.
Core Functionality	Uses large datasets to learn patterns and create new, original content based on those learned patterns.	Analyzes the environment, makes decisions, and adapts actions to meet goals. It's focused on completing tasks efficiently.
Interaction Style	Generally collaborative, as it works based on prompts, instructions, and input from users to generate content.	Fully autonomous; once set up, it acts on its own and doesn't need continuous interaction with humans.

Strengths	Excellent at creativity, content automation, and enhancing human innovation by producing large amounts of content.	Known for its efficiency, autonomy, and ability to scale in complex, dynamic environments that require decision-making.
Limitations	Dependent on the quality and range of training data; can produce biased or nonsensical outputs in some cases.	Complex to implement and requires careful ethical considerations and safeguards to ensure it makes appropriate decisions.

5. What is OpenAI Agents SDK?

The OpenAI Agents SDK is a software toolkit created by OpenAI. It helps developers build and manage multiple AI agents that work together to complete tasks.

Why use the Agents SDK?

The SDK has two driving design principles:

1. Enough features to be worth using, but few enough primitives to make it quick to learn.
2. Works great out of the box, but you can customize exactly what happens.

Main Features of OpenAI Agents SDK:

- **Agent loop:** Built-in agent loop that handles calling tools, sending results to the LLM, and looping until the LLM is done.
- **Python-first:** Use built-in language features to orchestrate and chain agents, rather than needing to learn new abstractions.
- **Handoffs:** A powerful feature to coordinate and delegate between multiple agents.
- Guardrails: Run input validations and checks in parallel to your agents, breaking early if the checks fail.
- **Function tools:** Turn any Python function into a tool, with automatic schema generation and Pedantic-powered validation.
- **Tracing:** Built-in tracing that lets you visualize, debug and monitor your workflows, as well as use the OpenAI suite of evaluation, fine-tuning and distillation tools.

What's the benefit of OpenAI agents?

The OpenAI Agents SDK empowers developers to create intelligent, scalable AI systems with ease. By providing a Python-first framework for building agentic workflows, the SDK simplifies development, enhances multi-agent coordination, and ensures safety through built-in guardrails.

References:

https://www.assemblyai.com/blog/llm-use-cases

https://research.ibm.com/blog/what-is-generative-AI

https://www.simplilearn.com/agentic-ai-vs-generative-ai-article

https://openai.github.io/openai-agents-python

https://humanloop.com/blog/openai-agents-sdk