[Helicone](https://www.helicone.ai/)

[Helicone Docs](https://docs.helicone.ai/getting-started/quick-start)

[GitHub Links](https://github.com/aishwaryanr/awesome-generative-ai-guide?tab=readme-ov-file#book-list-of-free-genai-courses)

* Applied LLMs Mastery full course content has been released!!! ([Click Here](https://github.com/aishwaryanr/awesome-generative-ai-guide/blob/main/free_courses/Applied_LLMs_Mastery_2024))
* 5-day roadmap to learn LLM foundations out now! ([Click Here](https://github.com/aishwaryanr/awesome-generative-ai-guide/blob/main/resources/genai_roadmap.md))
* [5-day LLM foundations roadmap](https://github.com/aishwaryanr/awesome-generative-ai-guide/blob/main/resources/genai_roadmap.md)
* [5-day LLM agents roadmap](https://github.com/aishwaryanr/awesome-generative-ai-guide/blob/main/resources/agents_roadmap.md)
* [Introduction to MM LLMs](https://github.com/aishwaryanr/awesome-generative-ai-guide/blob/main/resources/mm_llms_guide.md)
* [LLM Lingo Series: Commonly used LLM terms and their easy-to-understand definitions](https://github.com/aishwaryanr/awesome-generative-ai-guide/blob/main/resources/llm_lingo)
* [Large Language Models](https://rycolab.io/classes/llm-s23/) by ETH Zurich
* [Understanding Large Language Models](https://www.cs.princeton.edu/courses/archive/fall22/cos597G/) by Princeton
* [Introduction to Large Language Models](https://www.cloudskillsboost.google/paths/118/course_templates/539) by Google Cloud
* [1 Hour Introduction to LLM (Large Language Models)](https://www.youtube.com/watch?v=xu5_kka-suc) by WeCloudData
* [LLM Foundation Models from the Ground Up | Primer](https://www.youtube.com/watch?v=W0c7jQezTDw&list=PLTPXxbhUt-YWjMCDahwdVye8HW69p5NYS) by Databricks

Building LLM Applications

* [LLMOps: Building Real-World Applications With Large Language Models](https://www.udacity.com/course/building-real-world-applications-with-large-language-models--cd13455) by Udacity
* [Full Stack LLM Bootcamp](https://fullstackdeeplearning.com/llm-bootcamp/) by FSDL
* [Generative AI for beginners](https://github.com/microsoft/generative-ai-for-beginners/tree/main) by Microsoft
* [Large Language Models: Application through Production](https://www.edx.org/learn/computer-science/databricks-large-language-models-application-through-production) by Databricks
* [Generative AI Foundations](https://www.youtube.com/watch?v=oYm66fHqHUM&list=PLhr1KZpdzukf-xb0lmiU3G89GJXaDbAIF) by AWS
* [Introduction to Generative AI Community Course](https://www.youtube.com/watch?v=ajWheP8ZD70&list=PLmQAMKHKeLZ-iTT-E2kK9uePrJ1Xua9VL) by ineuron
* [LLM University](https://docs.cohere.com/docs/llmu) by Cohere
* [LLM Learning Lab](https://lightning.ai/pages/llm-learning-lab/) by Lightning AI
* [LangChain for LLM Application Development](https://learn.deeplearning.ai/login?redirect_course=langchain&callbackUrl=https%3A%2F%2Flearn.deeplearning.ai%2Fcourses%2Flangchain) by Deeplearning.AI
* [LLMOps](https://learn.deeplearning.ai/llmops) by DeepLearning.AI
* [Automated Testing for LLMOps](https://learn.deeplearning.ai/automated-testing-llmops) by DeepLearning.AI
* [Building Generative AI Applications Using Amazon Bedrock](https://explore.skillbuilder.aws/learn/course/external/view/elearning/17904/building-generative-ai-applications-using-amazon-bedrock-aws-digital-training) by AWS
* [Efficiently Serving LLMs](https://learn.deeplearning.ai/courses/efficiently-serving-llms/lesson/1/introduction) by DeepLearning.AI
* [Building Systems with the ChatGPT API](https://www.deeplearning.ai/short-courses/building-systems-with-chatgpt/) by DeepLearning.AI
* [Serverless LLM apps with Amazon Bedrock](https://www.deeplearning.ai/short-courses/serverless-llm-apps-amazon-bedrock/) by DeepLearning.AI
* [Building Applications with Vector Databases](https://www.deeplearning.ai/short-courses/building-applications-vector-databases/) by DeepLearning.AI
* [Automated Testing for LLMOps](https://www.deeplearning.ai/short-courses/automated-testing-llmops/) by DeepLearning.AI
* [Build LLM Apps with LangChain.js](https://www.deeplearning.ai/short-courses/build-llm-apps-with-langchain-js/) by DeepLearning.AI
* [Advanced Retrieval for AI with Chroma](https://www.deeplearning.ai/short-courses/advanced-retrieval-for-ai/) by DeepLearning.AI
* [Operationalizing LLMs on Azure](https://www.coursera.org/learn/llmops-azure) by Coursera
* [Generative AI Full Course – Gemini Pro, OpenAI, Llama, Langchain, Pinecone, Vector Databases & More](https://www.youtube.com/watch?v=mEsleV16qdo) by freeCodeCamp.org
* [Training & Fine-Tuning LLMs for Production](https://learn.activeloop.ai/courses/llms) by Activeloop

key factors to consider when selecting a Large Language Model (LLM). It highlights the technical specifications, performance metrics, and operational considerations that influence the choice of an LLM.

**Here's a breakdown of the key factors:**

**Technical Specs:**

* **Parameter Size:** This refers to the number of parameters in the model, which directly impacts its complexity and capabilities. Larger models typically have better performance but require more computational resources.
* **Context Window:** This represents the amount of text the model can process and reference in a single input. A larger context window allows the model to understand and generate more comprehensive and coherent responses.
* **Architecture:** The underlying architecture of the LLM, such as Transformer-based models, determines its learning capacity and efficiency.
* **Training Data:** The quality and quantity of the data used to train the model significantly influence its performance and ability to generalize to new tasks.

**Performance:**

* **Inference Speed:** This refers to how quickly the model can generate outputs given a prompt. Faster inference speeds are desirable for real-time applications.
* **Accuracy:** This measures the correctness and relevance of the model's outputs. High accuracy is crucial for tasks that require precise and factual information.
* **Reliability:** This refers to the consistency and dependability of the model's outputs. A reliable model should produce similar results for similar inputs.
* **Consistency:** This measures the coherence and logical flow of the model's generated text. A consistent model should generate text that is well-structured and free of contradictions.

**Operational Considerations:**

* **Cost:** The cost of deploying and running an LLM depends on factors like the model size, hardware requirements, and API usage.
* **Scalability:** The ability of the model to handle increasing workloads and adapt to changing requirements is crucial for large-scale applications.

**LLM Selection:**

The image also includes a selection of popular LLMs, such as LLaMA and Gemini, which can be evaluated based on these key factors to choose the most suitable model for a specific application.

LLM key factors

**LLM Key Factors**

The infographic presents a comprehensive overview of the key factors to consider when selecting a Large Language Model (LLM). These factors are categorized into three main areas:

**Technical Specs**

* **Model Dimensions:** This refers to the size and complexity of the LLM, measured in terms of parameters. Larger models typically have greater capacity and can handle more complex tasks.
* **Processing Scope:** The range of tasks and applications that the LLM is capable of handling.
* **Framework Design:** The underlying architecture and algorithms used to build the LLM.
* **Data Sources:** The quality and quantity of data used to train the LLM.

**Performance**

* **Execution Speed:** The speed at which the LLM can generate outputs.
* **Precision Level:** The accuracy and correctness of the generated outputs.
* **Stability:** The consistency and reliability of the LLM's performance over time.
* **Uniformity:** The coherence and logical flow of the generated text.

**Operational Considerations**

* **Expenses:** The cost of deploying and running the LLM, including hardware, software, and maintenance.
* **Expandability:** The ability to scale the LLM to handle increasing workloads and adapt to new requirements.

**LLM Selection**

The infographic also emphasizes the importance of carefully selecting an LLM that aligns with your specific needs and constraints. Key factors to consider include:

* **Technical Specs:** The LLM should have the necessary complexity and capabilities to handle your tasks.
* **Performance:** The LLM should generate accurate, consistent, and efficient outputs.
* **Operational Considerations:** The LLM should be cost-effective and scalable to meet your evolving requirements.

**In Summary**

The infographic provides a valuable framework for evaluating LLM options and making informed decisions. By considering these key factors, you can select an LLM that best meets your specific needs and drives the success of your AI initiatives.