**The LLM Landscape**

**Classifications and examples of LLMs**

We can separate LLMs into three different classifications.

**Architecture-Based LLMs**

* **Autoregressive models:** Generate text by predicting the next token in a sequence based on the previous tokens.
* Use a probability distribution to choose the next word or character
* Excel at generating fluent and contextually appropriate text
* Examples include the **OpenAI GPT series** and **Google Gemini**
* **Autoencoding models:** Grasp the context of words in a sentence by predicting masked tokens
* Trained by masking some input tokens and the model then predicting these masked tokens using the context
* Strong in tasks that need a deep understanding of context and semantics
* Examples include **BERT**, **RoBERTa** and **ELECTRA**
* **Sequence-to-sequence (Seq2Seq) models:** Compute tasks where the input and output are sequences e.g. translation, summarisation and text generation
* Typically consist of an encoder and a decoder where the encoder processes the input sequence and the decoder produces the output sequence
* Great for tasks that consist of transforming one type of text into another so for example, translating languages or generating summaries
* Examples such as **T5** and **BART**

**Availability-Based LLMs**

* **Open-Source models:** Freely available for anyone to use, modify and distribute
* Usually have a community of developers and researchers who offer support and contribute to development
* Offer transparency, flexibility and especially the ability to customise the model for specific needs
* Examples of such models are **LLaMA**, **BLOOM** and **Falcon**
* **Proprietary models:** Developed and maintained by private organisations and are available through licenses or subscriptions
* Demonstrate high performance and robustness
* Users receive dedicated support and updates from the provider
* Examples include **GPT-4**, **PaLM** and **Claude**

**Domain-Specific LLMs:** Tailored for particular industries or fields

* Examples are **BloombergGPT** for finance, **Med-PaLM 2** for healthcare and **Paxton AI** for legal

<https://labelyourdata.com/articles/types-of-llms>

<https://thedatascientist.com/5-best-examples-of-domain-specific-llms-in-ai/>

**Key Terminology**

* Slop: Poor quality and often unwanted AI generated content
* Hallucinations: Occurrences where LLMs generate factually inaccurate or illogical answers
* Token: Smallest unit of text that an AI model processes and understands
* Context Window: Number of tokens an LLM can process at once
* Knowledge Cutoff: Date when the LLM stopped learning from data
* Emergent behaviour: Unexpected skills or behaviours that arise in LLMs
* Pre-training: Initial training phase on large datasets to learn general patterns
* Fine-tuning: Adjusting a pretrained model on specific data for a specialised task
* Bias: Unintended prejudices learned from biased training data
* Prompt engineering: Practice of designing effective prompts to guide an AI model’s output

<https://whatisai.co.uk/what-is-ai-slop-understanding-the-risks/>

<https://medium.com/@stalin.t/llms-key-terms-explained-for-beginners-f1f17f7a1daf>

<https://mitsloanedtech.mit.edu/ai/basics/glossary/#Context_window>

<https://education.civitai.com/llm-glossary/>

**Resources**

* LLMs course to gain foundational knowledge: <https://trailhead.salesforce.com/content/learn/modules/large-language-models>
* Article on LLMs by IBM: <https://www.ibm.com/think/topics/large-language-models>
* Brief introduction to LLMs: <https://youtu.be/osKyvYJ3PRM?si=ju-x3qV1veLa4i1w>
* Dangers of LLMs: <https://www.nature.com/articles/d41586-025-03222-1>
* <https://www.anthropic.com/research/small-samples-poison>
* <https://youtu.be/7-UzV9AZKeU?si=MMaLPmwktaEJVoyG>
* <https://youtu.be/4__gg83s_Do?si=wJtCaJvgowLN_KXR>