Module 19: Foundations of Generative AI and Large Language Models (LLMs)

Learning outcomes:

1. Evaluate how humans and machines differ in their ability to understand language, drawing on historical foundations and modern LLMs.
2. Analyse the implications of model scale and architecture on performance, accessibility, and global AI development.
3. Apply the concept of emergence to prompt engineering to explore how user input shapes model behaviour.
4. Explain how transformers function, including attention mechanisms, tokenisation, and parameter complexity.
5. Synthesise foundational concepts of large language models, including emergence, transformers and hyperparameters, to evaluate their broader capabilities and limitations.

Large language models (LLMs)

LLMs are deep learning systems built on billions of parameters, trained to predict and generate text. Unlike humans, who use context and lived experience to understand language, LLMs rely on statistical learning from massive data sets. Their scale enables impressive fluency but also raises questions about whether they truly ‘understand’ language or simply mimic it.

Model size

- Defined by the number of parameters (e.g. GPT-2: 1.5B; GPT-3: 175B).
- More parameters = richer patterns, but also higher compute and storage demands.
- Precision formats (float32, float16, bfloat16) affect memory requirements.
- Larger models are more capable, but they increase cost, energy use and accessibility challenges.

Emergence and prompting

As models scale, they develop unexpected skills – a phenomenon known as emergence. Abilities such as translation, reasoning and summarisation appear without explicit training. Prompt engineering helps unlock these abilities by shaping input text to guide outputs.

Prompting strategies

- **Zero-shot:** no examples, direct instruction.
- **Few-shot:** provide a handful of examples.
- **Role prompting:** assign the model a persona.
- **Delimiting context:** mark input clearly with sections.
- **Step-by-step reasoning:** encourage structured thinking.

Prompting makes emergent capabilities more reliable, helping users balance creativity with control.

Transformers and attention

Transformers replaced sequential models (RNNs and LSTMs) with parallel processing, enabling efficient handling of long-range dependencies. Their key innovation is the **attention mechanism**, which lets each token in a sequence consider all others when forming context.

Core ideas

- **Parameters** are the learned values adjusted during training.
- **Hyperparameters** are the preset controls (e.g. learning rate, batch size and dropout rate) that affect stability and robustness.
- **Tokenisation** breaks text into subwords or characters for processing.
- **In attention (Q, K, V)**: queries (Q) seek information, keys (K) signal what's available and values (V) carry the actual content.
- **Scaling** is attention that requires N^2 comparisons, making long sequences costly.
- **Multi-head attention** captures different relationships in parallel.
- **Positional encoding** adds order information to tokens processed in parallel.

Applications and implications

LLMs underpin today's generative AI systems, enabling chatbots, translation, summarisation and code generation. But scale brings trade-offs: enormous training costs, environmental impact and questions of accessibility and fairness. Understanding parameters, prompting and attention is key to both leveraging and critiquing these technologies.

Tip: Bigger isn't always better. Consider task requirements, resource constraints and ethical implications before assuming the largest model is the right choice.