LLMs for Dummies

My first question is simple... what is a Large Language Model (**LLM**)? I'd say most people could have a good stab at that question and their answer would most likely include the words *ChatGPT* or *Gemini*. I'm of course aware of the household chatbots and understand on the surface how they operate but I'm curious as to how this revolutionary subset of AI functions under the hood. How did it all begin? What were the key breakthroughs that allowed this technology to turn into what we know it as today?

This is the first topic want to delve into as it currently seems to be the most popular form of AI, and I'm sure it's something that a lot of people can relate to. These chatbots have become part of everyday life for a lot of people but many, including myself, don't even know how they work (a poor confession from a Computer Science student).

My queries going into this include what its origins are, how they operate at a fundamental level and what the future of LLMs look like, as well as any possible pitfalls.

Origins

As expected, the LLM is all about language. More specifically, it focuses on the patterns of meaning in language and bridges the gap between computation and human communication.

The start of LLMs can be seen to go all the way back to 1883 with French philologist Michel Bréal. As well as being the creator of the marathon race as we know it, he is also recognised as the founder of modern semantics (the study of meaning in language). In 1916, Swiss linguist Ferdinand de Saussure's Cours de Linguistique Générale (*Course of General Linguistics*) was posthumously published which laid the groundwork for Natural Language Processing (**NLP**) (De Saussure, 1959). The aim of NLP is to translate human communication into a way that computers can understand, and then back again. It's a field of study that sees languages as functional systems, as opposed to cultural expression.

After WWII, the want for language translation machines was prevalent, though as expected proved very difficult to implement. However, mathematics was already a universal language and proved great bedrock for these machines to be built on.

Funding for this language-based technology ebbed and flowed and in 1966, Joseph Weizenbaum created **ELIZA**, a computational psychiatrist and the first program seen as directly implementing NLP. This was a creation based off Weizenbaum's theory that communication between a human and machine was shallow, purely operating to reflect the human's input to create conversation. This led to a machine laden in question-based responses. Even though **ELIZA** didn't allow for in-depth, sophisticated conversation, users often felt very human emotions towards it. A web-based version of **ELIZA** can be used here.

By the 1980s, IBM had started creating *Small Language Models* (**SLM**). These were trained on small sets of data, namely transcribed speech, and could complete limited text prediction and were in the early stages of machine translation. Though not the big, powerful chatbots we use today, these machines gave us insight that this type of technology could be highly beneficial.

As compute power advanced rapidly, SLMs were able to be trained at much faster, more efficient rates, and when the internet came around, they suddenly had a vast, sophisticated dataset they could be openly trained on. By the 2010s, *Deep Learning* and other advances in surrounding technologies had allowed these language models to become so advanced that they gave us what are now known as *Large Language Models*. Developments in this field, such as the introduction of the transformer architecture, improved attention mechanisms and increased overall scale has allowed the LLM to truly flourish, giving us the household names such as **ChatGPT**, **Gemini**, **Claude**, and so on.

Basic Operation

LLMs use text prediction to generate responses prompted by user input. A solid understanding of how this works can be found in the the revolutionary paper Attention Is All You Need (Vaswani et al, 2017), or for those that favour visual learning, Grant Sanderson's (3Blue1Brown) video series on YouTube. Trying to simplify as much as possible, this is the basic operational process transformer-based LLMs go through to generate text (to my limited understanding at least):

1. **Tokenisation** -
   - Training data is fed into the model and broken down into *tokens* through a process known as tokenisation (tokens could be words, parts of words or even individual characters)
2. **Embedding** -
   - Each token is assigned a high dimensional vector that captures its basic semantic meaning
      - *Positional Encoding* - Positional information is added to the embeddings so the model is aware of the order of tokens
3. **Attention Block** -
   - Vectors are updated based on their relation to other tokens. This allows for:
      - *Dynamic Weighting* - Giving higher importance to tokens that are more relevant based on current context
      - *Long-range Dependency* - Allowing each token to affect all others
      - *Contextual Understanding* - Resolving ambiguities (e.g. homonyms) by using surrounding tokens
4. **Multilayer Perceptron** (**MLP**) -

- While the *attention block* handles relationships between tokens, the MLP processes the individual's token information
- It adjusts a token's vector value based on information said token gained from the *attention block*

5. **Repetition** -
   - Vectors are passed through the *attention* and MLP layers numerous times
   - This allows the model to build context-rich, intelligent language
6. **Output** -
   - After the final repetition, a vector of *logits* is produced for each possible next token
   - A softmax function turns these logits into a normalised probability distribution
   - The token with the highest probability will *typically* be chosen as the next output token

As previously stated, this is a step-by-step operation for a *transformer* based LLM. Google's 2017 paper Attention Is All You Need (Vaswani et al, 2017) outlined this architecture and created a new standard within the LLM community. Before this breakthrough, architectures for LLMs included *Recurrent Neural Networks* (**RNN**) and *Convolutional Neural Networks* (**CNN**). RNN is sequential in processing meaning it can only handle one word at a time (tokens don't have global effect on future predictions) and CNNs struggled to understand order. The transformer-based architecture solved these issues by incorporating both its *attention block* and MLP.

One final concept on the fundamental operation of how LLMs operate is *distillation*. This is a practice where the core performance of an LLM is extracted and simplified so it can perform as the basis of a completely new language model. This allows developers to get the same results from enormous models at a much smaller price point - something OpenAI accused DeepSeek of at the start of this year.

The Future of LLMs & Possible Downfalls

Like any form of AI, LLMs will only become more powerful as we go on. More companies will use chatbots as part of their platform to 'enhance' customer services, law firms will incorporate its uses to speed up monotonous drafting tasks (Hadi et al., 2024) and inboxes will continue to categorise emails based on its contents. LLMs are used extensively in the world of code generation, one of the many contributions that gave them life in the first place. They're already writing code for other AI systems, and this is expected to play an increasingly significant role in their development. Given this notion, there's every chance that the next major LLM will be trained and tested by a fellow LLM.

I've spoken grandly of this form of AI for some time now, so I'd feel it right to cover a couple pitfalls and rooms for improvement, so to speak. An area LLMs still need to better in is literature review research. Due to its operation simply being enhanced text prediction, if one were to automate tasks such as identifying relevant written material and summarising content, the model could in theory produce fake references and provide sources that don't exist. This could lead to mass misinformation and incorrect conclusions which could be harmful, especially in the scientific community.

One final point on LLMs is their environmental impact. Though their uses are vast and impressive, the amount of water and energy they require to operate is large. Like all major forms of AI, the water necessary to cool data centres is shocking. For example, DeepSeek-R1 uses 150ml of water per query. This doesn't seem like a large sum in isolation but across a year the impact is substantial. Going to another model, it's estimated that across a year, someone using GPT-4o at an average rate of 8 queries per day could use between 1,334,991kL and 1,579,680kL of water (Jegham et al., 2025). You may be thinking that though this does seem a lot, it's okay because water can be recycled. However, this is not the case. These statistics refer to water that's evaporated during cooling and permanently removed from freshwater ecosystems.

Though this technology has vast history and is by all means revolutionary, going forward I hope that more efficient ways of operation are developed to get the same performance out of LLMs without its excess in overall consumption. Who knows what it will produce next, only we as the user can prompt that.

1. De Saussure, F., 1959. **Course in General Linguistics**. Translated by W. Baskin. New York: Philosophical Library.
2. Jegham, N., Abdelatti, M., Elmoubarki, L. and Hendawi, A., 2025. **How Hungry is AI? Benchmarking Energy, Water, and Carbon Footprint of LLM Inference**. Available at: https://arxiv.org/html/2505.09598v1#S6
3. Hadi, U., Tashi, Q., Qureshi, R., Shah, A., Muneer, A., Irfan, M., Zafar, A., Shaikh, M., Akhtar, N., Hassan, S., Shoman, M., Wu, J., Mirjalili, S. and Shah, M., 2024. **LLMs: A Comprehensive Survey of Applications, Challenges, Datasets, Limitations, and Future Prospects**. Available at: https://www.techrxiv.org/users/618307/articles/682263-large-language-models-a-comprehensive-survey-of-its-applications-challenges-limitations-and-future-prospects
4. Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A.N., Kaiser, L. and Polosukhin, I. (2017). **Attention Is All You Need**. Cornell University. Available at: https://arxiv.org/abs/1706.03762.