Houzair Koussa

Software Engineer

Maintainer @ thephilosophicalcode.com

LLM, Language And Meaning

About

Overview of a theory of language* conceptually inspired by recent developments in LLMs**

*A theory about the nature of language

**Mostly Generative Pre-trained Transformers (GPTs)

Preliminary Notes

Inspired by the ideas and ongoing work of Professor Elan Barenholtz (Florida Atlantic University)

Appeal to charity and scepticism

Slides and references @ github.com/houzyk/talks

Easter egg in the title “LLM, Language And Meaning” (hint - PHP)

Agenda

1. How LLMs Conceptually Work

2. The LLM Theory Of Language

3. Ramifications

4. Challenges

5. Q & A

How LLMs Conceptually Work

Overview

Given a prompt, an LLM generates an output that semantically follows the prompt

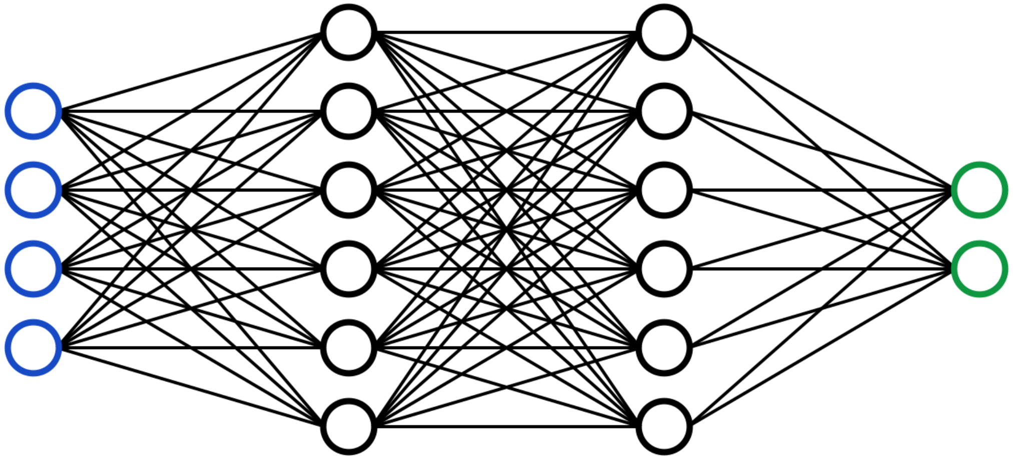
Major LLMs are implemented using GPTs (see *Attention Is All You Need*)

Conceptual workings of LLMs ≠ Implementation details of LLMs

How LLMs Conceptually Work

Overview

An LLM is a high-dimensional network of matrix-like nodes and weights

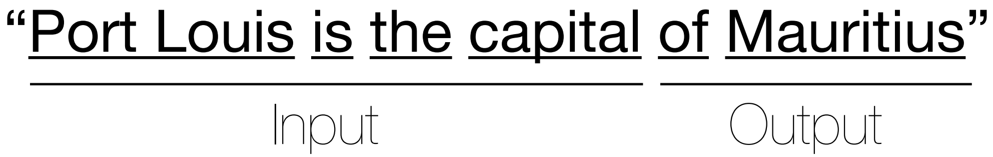


Conceptually, we'll call this permutation of nodes and weights (aka "layers") as an LLM's *embedded space*

How LLMs Conceptually Work

Training

A massive sanitised text corpus is tokenised and labelled



“Port Louis is the capital of Mauritius”

Input

Output

The labelled corpus is prompted into a randomised embedded space

The LLM outputs a probability distribution of next tokens which semantically follow the prompt

How LLMs Conceptually Work

Training

A cost-function compares expected next token probabilities against the distribution

For example, given “Port Louis is the capital”, P(“of”) may be .2 instead of .9

Back-propagation readjusts the embedded space optimising for cost

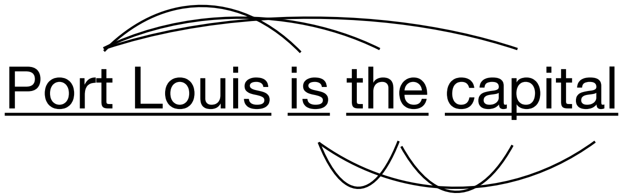
Iteratively, the embedded space is moulded to semantically follow the corpus

Conceptually, the embedded space *encodes meaning*

How LLMs Conceptually Work

Generation

When prompted, a GPT’s attention mechanism kicks in - every token “informs” each other of their context



Port Louis is the capital

Conceptually, a prompt is imbued with meaning from the embedded space

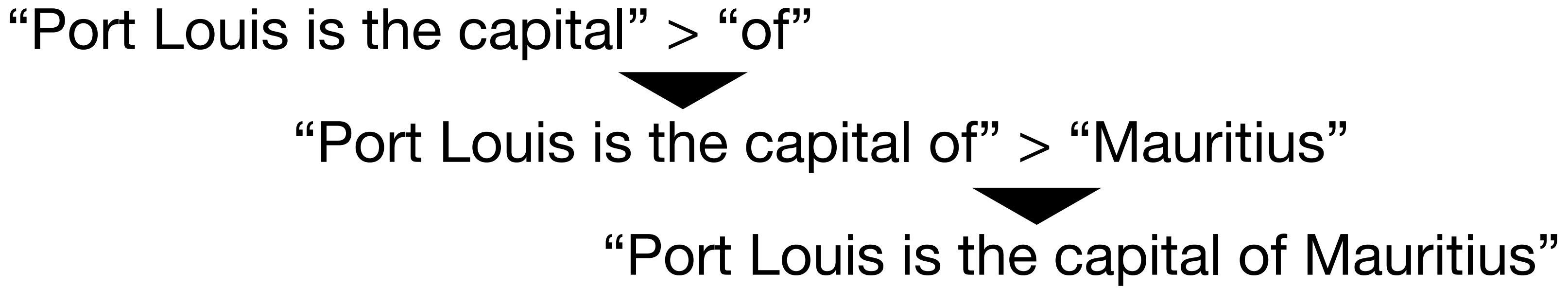
As previously mentioned, the LLM outputs a probability distribution of next tokens

How LLMs Conceptually Work

Generation

A next token is sampled from the distribution

Auto-regression - the next token is recursively appended to the prompt and re-prompted into the LLM



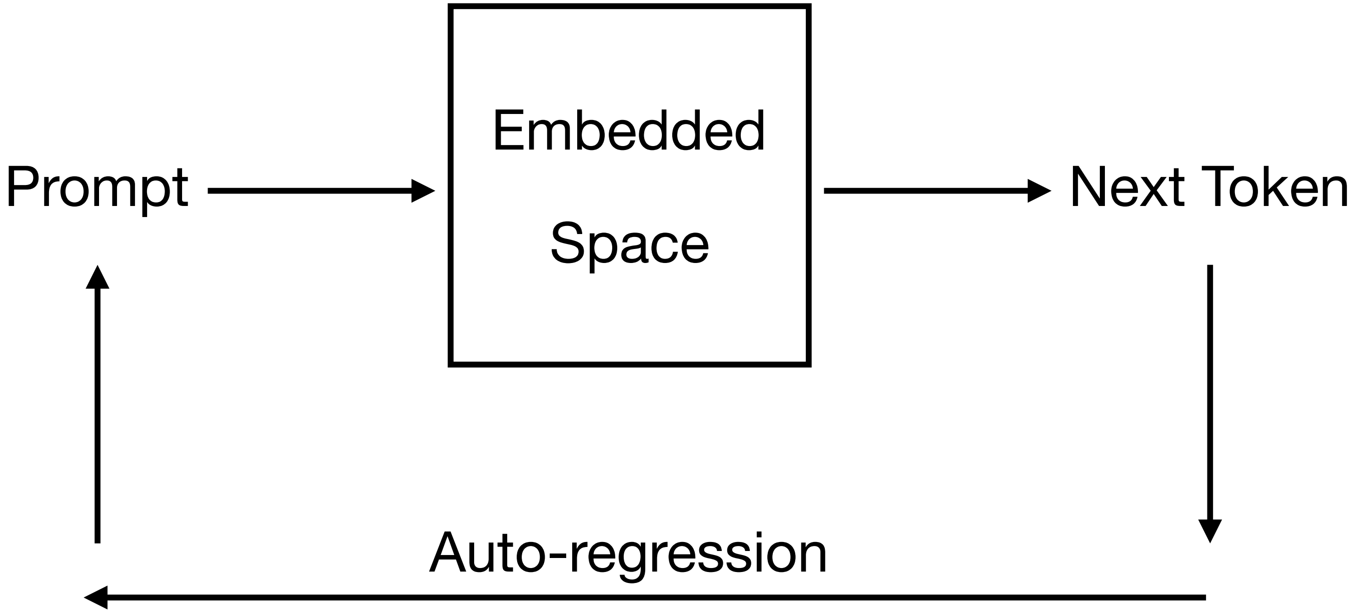
“Port Louis is the capital” > “of”

“Port Louis is the capital of” > “Mauritius”

“Port Louis is the capital of Mauritius”

How LLMs Conceptually Work

In A Nutshell



Prompt

Embedded Space

Next Token

Auto-regression

The LLM Theory Of Language

Key Observation And Philosophical Question

Observation - LLMs can generate meaningful sentences without knowing/ understanding the meaning of these sentences

LLMs don’t need to know what the word “Mauritius” means to use it. It doesn’t know Mauritius itself

Question - *why* do LLMs do that? (we already know the “how”)

The LLM Theory Of Language

A Tentative Answer

LLM training data is massive. LLMs found deep linguistic patterns and are merely replicating these patterns

This answer isn't sufficient. It doesn't *particularly* tell us why an LLM, *and not any other system,* is so good at replicating these patterns

Maybe our tentative answer is only a consequence which we're mistaking for the actual root cause

The LLM Theory Of Language

Professor Elan Barentholtz's Answer

LLMs found deep linguistic patterns because they discovered an *essential property of language* itself. LLMs have modelled language

Language is *conceptually like* an LLM!

So, LLMs are really good at generating meaningful sentences because they found and replicated deep linguistic patterns *produced by another LLM (language)*

The LLM Theory Of Language

Core Idea

Language is essentially an independent auto-regressive next-token-predicting system

The LLM Theory Of Language

Caveats

This theory is *only* about the nature of language

We’re not denying subjectivity, consciousness, qualia or free-will

Language is conceptually like an LLM - we’re not saying that it shares the same implementation details as current LLMs

We’re not saying that we have neural networks or transformers in our brain

The LLM Theory Of Language

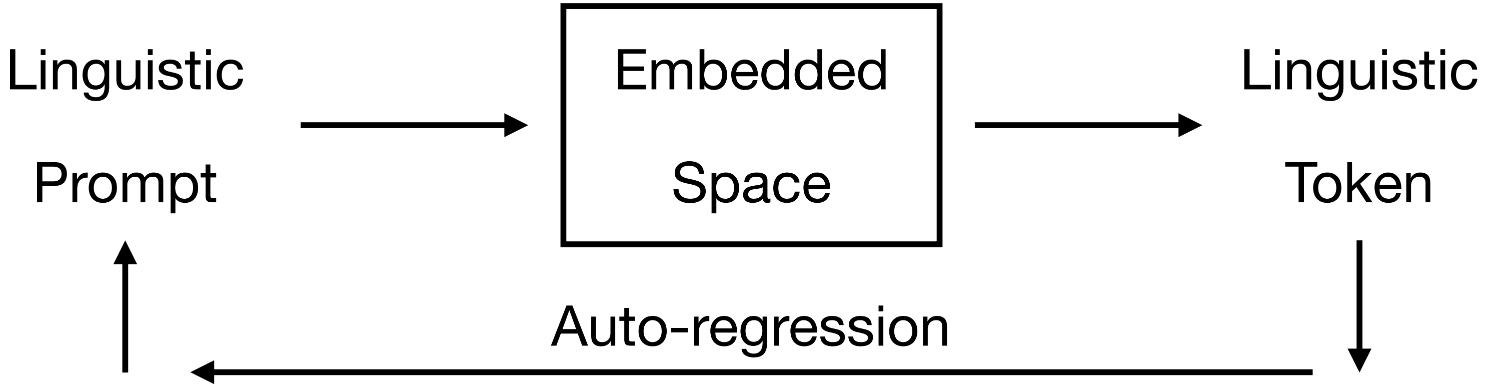
Core Concepts

Given a prompt, language generates next linguistic tokens

Embedded space - encodes meaning and relations between linguistic tokens

Auto-regression - language is fundamentally recursive

Independent - language is a closed system



Ramifications

Conversation Is Just Prompt Exchange

Imagine that Alice and Bob are having a conversation

- Alice starts talking. Alice generates a prompt for Bob’s LLM
- Bob’s LLM produces Alice’s reply which prompts Alice’s LLM back
- Repeat

Inner-monologue - Inner LLM prompting itself

PS: Involuntary thoughts. Tip-of-the-tongue phenomenon

Ramifications

Meaning Isn’t Needed

Language is independent of meaning

There is no extra-linguistic process (like reference) which attaches meaning to words

Instead, meaning is encoded in language’s embedded space

Intuitively, vectors in close proximity share similar meaning

So, the meaning of the word “Mauritius” is derived from its position and relations to other words in the space

Ramifications

Linguistic Memory Isn’t Needed Neither

Since a word’s meaning is already encoded in the embedded space, there is no explicit linguistic storage nor retrieval

When we use a word, language doesn’t pick-out some information from an extra-linguistic memory space. The word doesn’t even reside in memory

There is no explicit in-memory modelling of the world

Challenges

Meta-theory Notes

Key strengths -

- Parsimonious and simple
- Falsifiable

Even if falsified -

- The theory may be supplemented or augmented
- It may still act as a solid framework to ask further questions

Challenges

Diffusion

Briefly, diffusion models (like DALL-E) take *unstructured* noise as input. The noise is progressively enhanced until a desired output is reached

In contrast, an LLM takes *structured* linguistic tokens as input

Auto-regressive next-token prediction ≠ progressive noise enhancement

Falsifiability experiment - Achieve similar/better language generation with diffusion rather than auto-regressive next-token prediction

Challenges

Grounding Problems

Core issue - need to bridge language, as a closed system, to extra-linguistic phenomena

Explanatory gap - language attaches to actions, emotions and the world

Potential solution - shared embedded space

Language’s embedded space bridges to non-linguistic phenomena via this shared embedded space

Challenges

Grounding Problems

The linguistic embedding of “Mauritius” is *enriched* with the sense-experience embedding of Mauritius in the shared space

Importantly, this is not a simple “mapping” of linguistic to non-linguistic

Falsifiable predictions -

- Mapping everything onto language is a “dumb trick”
- Existence of a shared space -
    - Infer how a particular language sounds like only from how it’s written
    - Two *independent* LLMs successfully communicating

Challenges

Language Learning And Usage

Core issue - current LLMs do not reflect human language learning and usage

Current LLM’s embedded spaces are “fixed” and they don’t “forget” context

In contrast, we’re constantly learning and forgetting language

Potential but partial solution -

- In-context learning capabilities and fine-tuning
- Context decay function

Challenges

Miscellaneous

Infant language learning

Historical vocabulary growth

Indexicals like “there”, “here” or “now”

Animal language and communication

Key Takeaways

LLMs are auto-regressive next-token-predicting systems

They can generate meaningful sentences without understanding the meaning of these sentences

That’s because they encode patterns produced by another LLM - language

Language is an independent auto-regressive next-token-predicting system

“Words. But what are words, really, hmmm? They’re mere sounds with meanings dangling from them. That have no logic. They find their own way. Arising from the squabble between a sinking body and a drowning mind, they grab hold of antonyms. The seed planted was a date tree; what blossomed was hibiscus. They wrestle with themselves - wrapped up in their own game.”

Tomb of Sand (translated by Daisy Rockwell) - Geetanjali Shree