Linguistics & LLMs

A Preliminary Take

1

Chomsky's Conjecture (1957)

There is an innate *universal grammar* of human language.

All human languages share a common structural basis.

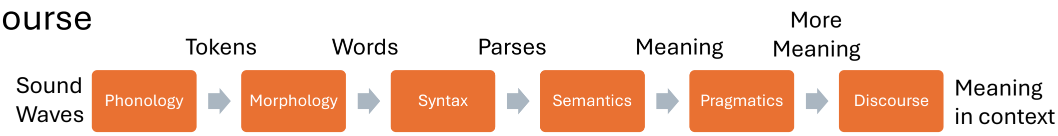
The ability for humans to acquire language is innate.

This has been a cornerstone in the development of linguistic models and has influenced the design of early NLP systems.

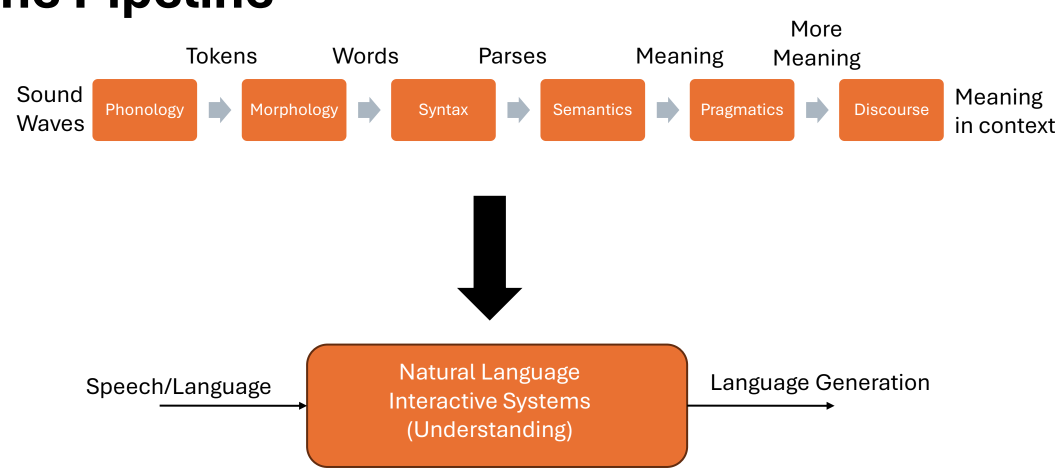
2

The Pipeline

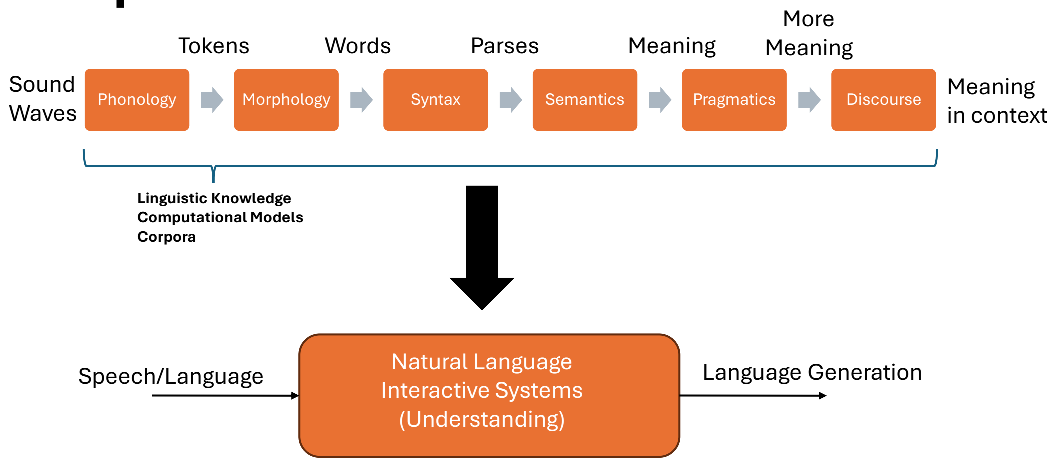
- Phonology
- Morphology
- Syntax
- Semantics
- Pragmatics
- Discourse



The Pipeline



The Pipeline

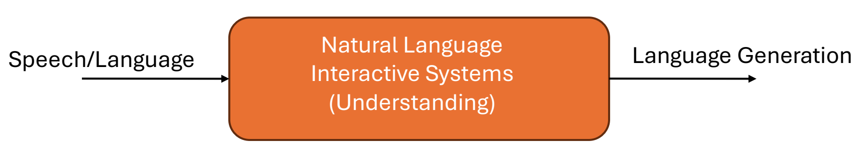


12/3/2024 5

5

Applications

- Semantic search engines
- E-Mail filters
- Sentiment analysis
- Chatbots
- Virtual Assistants
- Knowledge Management
- Machine Translation
- Text classification
- Text extraction
- Text summarization
- Predictive Text/Autocorrect
- Customer Feedback Analysis
- Customer Support
- Content Analysis (surveillance & security)
- Fraud Detection
- Etc.



6

Language Modeling

- The problem:

  Given a sequence of words $w_1, w_2, \ldots, w_{i-1}$

  Predict w_i

  where $w_1 \ldots w_i \in$ {<vocabulary of words>}
  i.e.

$$P(w_i | w_1, w_2, \ldots, w_{i-1})$$

- Example,

  Input: `the cat sat on the`
  Output: `the cat sat on the` **`mat`**
  (97%)

- Any system that that can do this prediction is called a **language model**.

- A language model is a probabilistic model of language.

7

7

Language Model: Word N-Gram Models

- $P(w_i | w_1, w_2, \ldots, w_{i-1})$ depends on i-1 previous words

  This is called an i-gram model.

  Unigram is single word frequencies
  Bigram is pair frequencies
  Trigram is 3-word frequencies

  Given: `He likes`
  Output: `He likes` **`being`**

- Large N-gram LLMs used for Machine translation (2005)

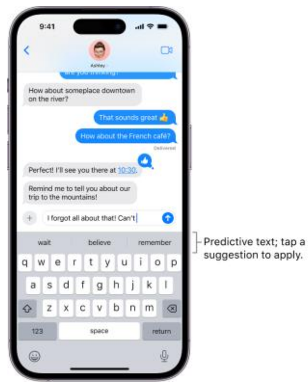
He likes attention	196
He likes bananas	51
He likes barbeque	44
He likes baseball	188
He likes basketball	57
He likes beer	281
He likes being	2026
He likes best	165
He likes better	55
He likes big	380
He likes bikes	47
He likes birds	111
He likes blue	42
He likes books	191
He likes both	276
He likes boys	90
He likes bread	73

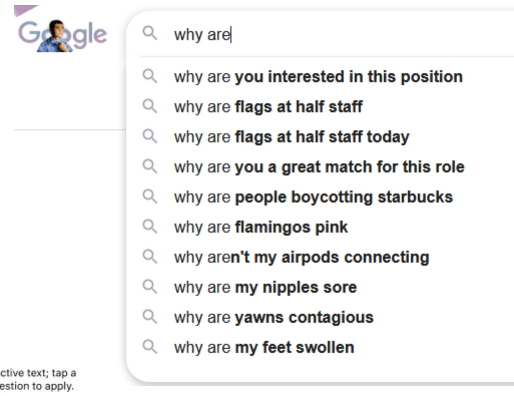
8

8

Applications

- Google Search
- Next word prediction in smart phone texts
- Writing assistants
- Etc.

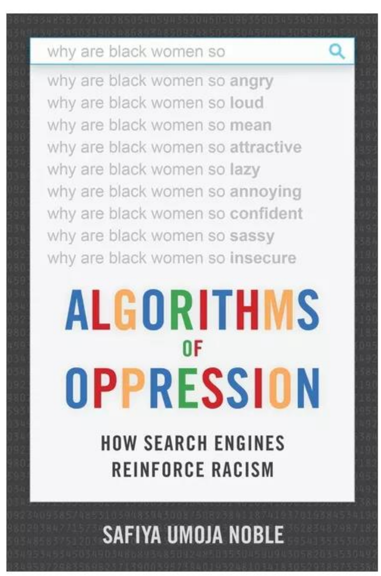




9

9

Issues



10

10

NNs for NLP Architectures

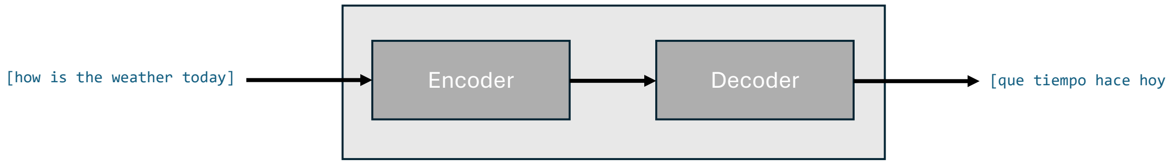
- **Representing words** and **word order** is important in NNs for NLP tasks.
- Representing words as vectors: One-hot encoding, Word2Vec, Word Embedding
- Inputing a word at a time ignores word ordering.
- **Transformers** track word order information and pay attention to different parts of a sentence.

Transformers, 2017

- **Sequence to sequence models** – processing words as a sequence

  Models learn their own features (like word embedding and word order) using raw word sequences.
- 2016-17, RNNs were all the rage for NN sequence models for NLP
- Transformers replaced many RNNs
  "*Attention is all you need*" by Vaswani, *et al*, 2017

Sequence to Sequence Models



- Given an input sequence of words (in English)

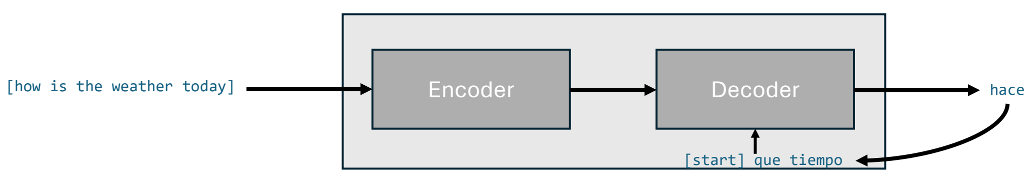
  Output a sequence of words (in Spanish)

- **Encoder-Decoder model**

  Encoder turns an input sequence into an intermediate representation.

  Decoder is trained to predict the next token (i) in the output sequence by looking at (1) Previous output sequence ($0..i-1$) and (2) the encoded input sequence.

Sequence to Sequence Models



- Given an input sequence of words (in English)

  Output a sequence of words (in Spanish)

- Training data: several sequences of input-output pairs

- **Encoder-Decoder model**

  Encoder turns an input sequence into an intermediate representation.

  Decoder is trained to predict the next token (i) in the output sequence by looking at (1) Previous output sequence ($0..i-1$) and (2) the encoded input sequence.

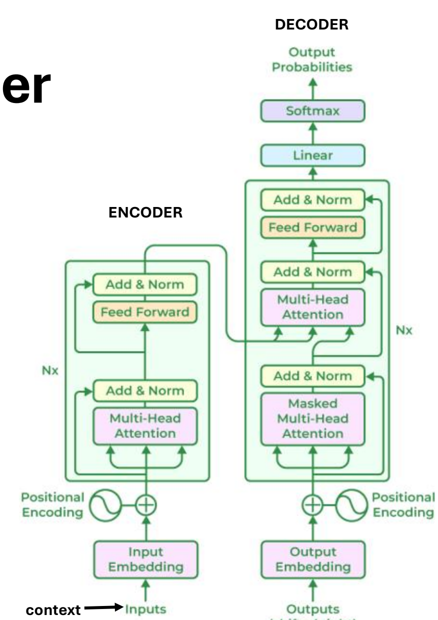
Transformers: Key Components/Ideas

- **Encoder-Decoder**
- **Positional Encoding**
  (preserves positional information in input sequence)
- **Attention Mechanism** (*aka* Neural Attention)

Transformer: Encoder+Decoder

- This is a full sequence to sequence transformer architecture.
- The encoder produces context aware representations of each input token.
- The decoder reads in $0..i-1$ tokens already produced and outputs the ith token.

  It uses neural attention to identify tokens in input sequence that may be closely related to the token it is trying to predict.



From: https://www.geeksforgeeks.org/large-language-model-llm/amp/

Transformer: Applications

- Can be used for any sequence-sequence task

  Machine Translation: Convert text in a source language into text in a target language.

  Text Summarization: Convert a long document into a shorter version that retains important information.

  Question Answering: Convert an input question into an answer.

  Chatbots: Convert a dialog prompt into a reply to this prompt, or convert a history of a conversation into the next reply in the conversation.

  Text Generation: Convert a text prompt into a paragraph that completes the prompt.

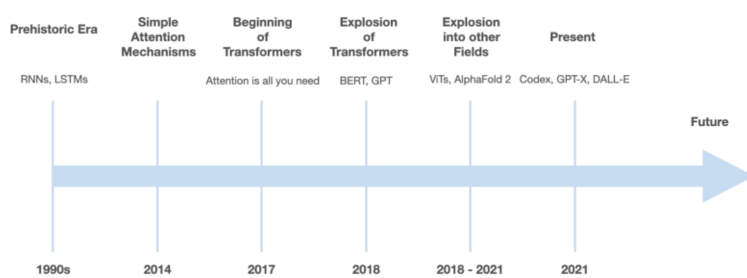
  Coding: Convert a text prompt into a program (or even complete application)

  etc.

17

17

A short History of Transformers

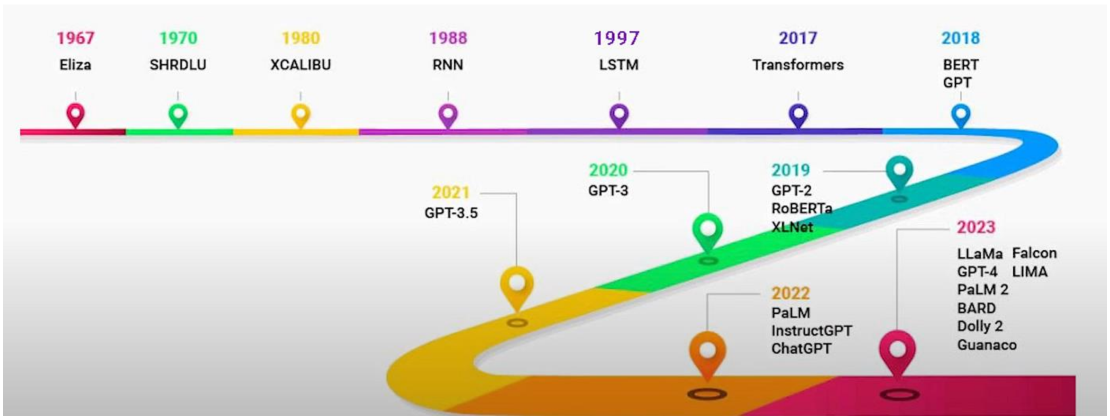


From: https://dair-ai.notion.site/Introduction-to-Transformers-4b869c9595b74f72b088e5f2793ece80

18

18

A More Current Picture



From: https://medium.com/@researchgraph/brief-introduction-to-the-history-of-large-language-models-llms-3c2efa517112

19

Large Language Models (LLMs)

- Trained on massive amounts of data (e.g. LLAMA-2 used 10TB)
- Involve billions of parameters (e.g. LLAMA-2 has 70 billion)
- Use large amounts of computational resources (e.g. LLAMA-2 used 6000 GPUs, took ~12 days, cost over $2million)
- Use tremendous amount of energy.

  Example LLMs: OpenAI's GPT models, Google's PaLM (used in Bard) and Gemini, Meta's LLaMa and BLOOM, Ernie 3.0 Titan, Anthropic's Claude.

20

20

LLMs Training

- **Pre-Training**
  Uses copious amounts of text (high-quality scrapped from the entire web).
  Text is huge, but low quality, raw. Results in a **Base Model**.

- **Fine Tuning**
  Uses smaller but high-quality domain specific text (e.g. human generated and labelled text/documents). Training on this text is built on top of the pre-trained transformer (**alignment**).
  The result is an **Assistant Model**. Cheaper, faster (takes ~ 1 day). Undergoes evaluation and incorrect responses are fixed (by humans, adding to training data).

- **Fine Tuning (RLHF)**
  Have the transformer generate multiple responses, humans select good candidate answers. This is called Reinforcement Learning with Human Feedback (RLHF).

- **Tool Integration**
  In the future, LLMs are being evolved into tool use capabilities. For example, a chat assistant that can draw plots by generating Python Matplotlib code, or doing web searches to get additional facts/data, generate code, order online items, etc.

21

21

Large Language Modeling

- The problem:

  Given a sequence of words $w_1, w_2, \ldots, w_{i-1}$

  Predict w_i

  where $w_1 \ldots w_i \in$ {<vocabulary of words>}
  i.e.

$$P(w_i | w_1, w_2, \ldots, w_{i-1})$$

- Example,

  Input: `the cat sat on the`
  Output: `the cat sat on the` **`mat`**
  (97%)

- The ***context*** ($w_1 \ldots w_{i-1}$) for Large Language models can be as high as 2000-100000 tokens.

22

22

Demo

23

LLMs Embody a Completely Different Approach

- Instead of finding Chomsky style universal rules (e.g., *subject comes before a verb and is followed by an object* in a sentence) that define a grammar, LLMs “discover” or “learn” grammar that is encoded in the weights of an optimized neural network.

- While LLMs seem to “memorize” the text they are trained on, they are also able to generalize to produce new text. This is an essential feature of NNs.

- LLMs exhibit wider and more powerful language capabilities: generating syntactically correct discourse, translation, answering questions, writing code, etc.

24

LLM Critics

- Stochastic Parrots/Internet Remixed

  LLMs essentially mimic and string together **probabilistic linguistic patterns** from their training data **without truly understanding** the meaning behind the words…how a parrot might imitate sounds without comprehending their significance.

25

Hallucinations

26

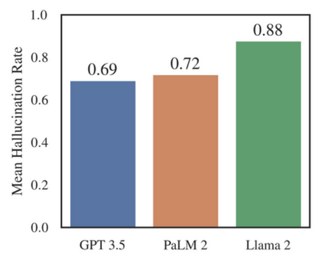
Law, Regulation, and Policy

Hallucinating Law: Legal Mistakes with Large Language Models are Pervasive

A new study finds disturbing and pervasive errors among three popular models on a wide range of legal tasks.

Jan 11, 2024 | Matthew Dahl, Varun Magesh, Mirac Suzgun, Daniel E. Ho

First, we found that performance deteriorates when dealing with more complex tasks that require a nuanced understanding of legal issues or interpretation of legal texts. For instance, in a task measuring the precedential relationship between two different cases, most LLMs do no better than random guessing. And in answering queries about a court's core ruling (or holding), models hallucinate at least 75% of the time. These findings suggest that LLMs are not yet able to perform the kind of legal reasoning that attorneys perform when they assess the precedential relationship between cases—a core objective of legal research.



Legal hallucination rates across three popular LLMs.

From: https://hai.stanford.edu/news/hallucinating-law-legal-mistakes-large-language-models-are-pervasive

27

In Summary…

- LLMs can process texts with extraordinary success. Often in a way indistinguishable from human output. They do this while lacking any intelligence, understanding or cognitive ability.

  At the same time…

- LLMs are brittle (susceptible to catastrophic failure)
- LLMs are unreliable (they output false or made-up information)
- Their reasoning prowess is rudimentary at best.

28

Responses to LLM Proponents

- "Unconstrained" Learning from Big Data is not human

  Despite the impressive performance of LLMs, humans achieve their capacity for language after exposure to **several orders of magnitude of less data**.

  Young children become fluent users of their native language with relatively little exposure (**innateness!**)

29

Simulation is not Duplication

- What can the artificial tell us about the natural?

  Akin to comparing airplanes to flying birds.

  Is human language faculty like LLM just because ChatGPT-4 outperforms most test takers on LSATs?

  What does a computer excelling at a human task tell us anything about how humans do the same thing? (Also, Chess!)

30

LLMs are not a Scientific Theory

- Prediction is not understanding

  Linguistic theory should provide explanations for linguistic capacities, not merely predict text.

  LLMs are a tool, not a theory.

  Linguistic theories offer explanations.

31

Linguistic Theories Make Fundamental Distinctions

"*Colorless green ideas sleep furiously.*" (Chomsky, 1957)

Shows how syntax is independent of semantics.

All bigrams in the sentence (*colorless green*, *green ideas*, *ideas sleep*, *sleep furiously*) make little or no sense. Yet, syntactically, it is a well-formed sentence. Just like:

"*Fluffy orange cats sleep peacefully.*"

Without linguistic theory, we do not know what distinctions we expect LLMs to make.

32

Why do LLMs excel linguistically?

- We do not yet know why LLMs show the behavior that they do. Any linguistic claims need to provide an explanation.

  (1) LLMs do it somehow.
  (2) How they do it is very different from the Chomsky approach.
  (3) The LLMs approach (what that is) works really well.

  Bigger question:

  How (if at all) should LLM technology influence linguistic theories?

33

A New Interdisciplinary Science of Language?

- Embrace the new ideas
- Rethink (some) old assumptions
- Integrate aspects of NN learning into the new theories

34

References

- N. Chomsky, *Syntactic Structures* (1957), Martino.
- S. T. Piantadosi, *Modern language models refute Chomsky's approach to language* (2023), https://doi.org/10.48550/arXiv.2308.03228
- E. bender, T. Gebru, A. MacMillan-Major, S. Shmitchell, *On the Dangers of Stochastic Parrots: Can Language Models be Too Big?* FAccT '21: Proceedings of the 2021 ACM Conference on Fairness, Accountability, and Transparency (2021), ACM Press.
- J. Kodner, S. Payne, J Heinz, *Why Linguistics Will Thrive in the 21st Century: A Reply to Piantadosi* (2023), https://doi.org/10.48550/arXiv.2308.03228

35