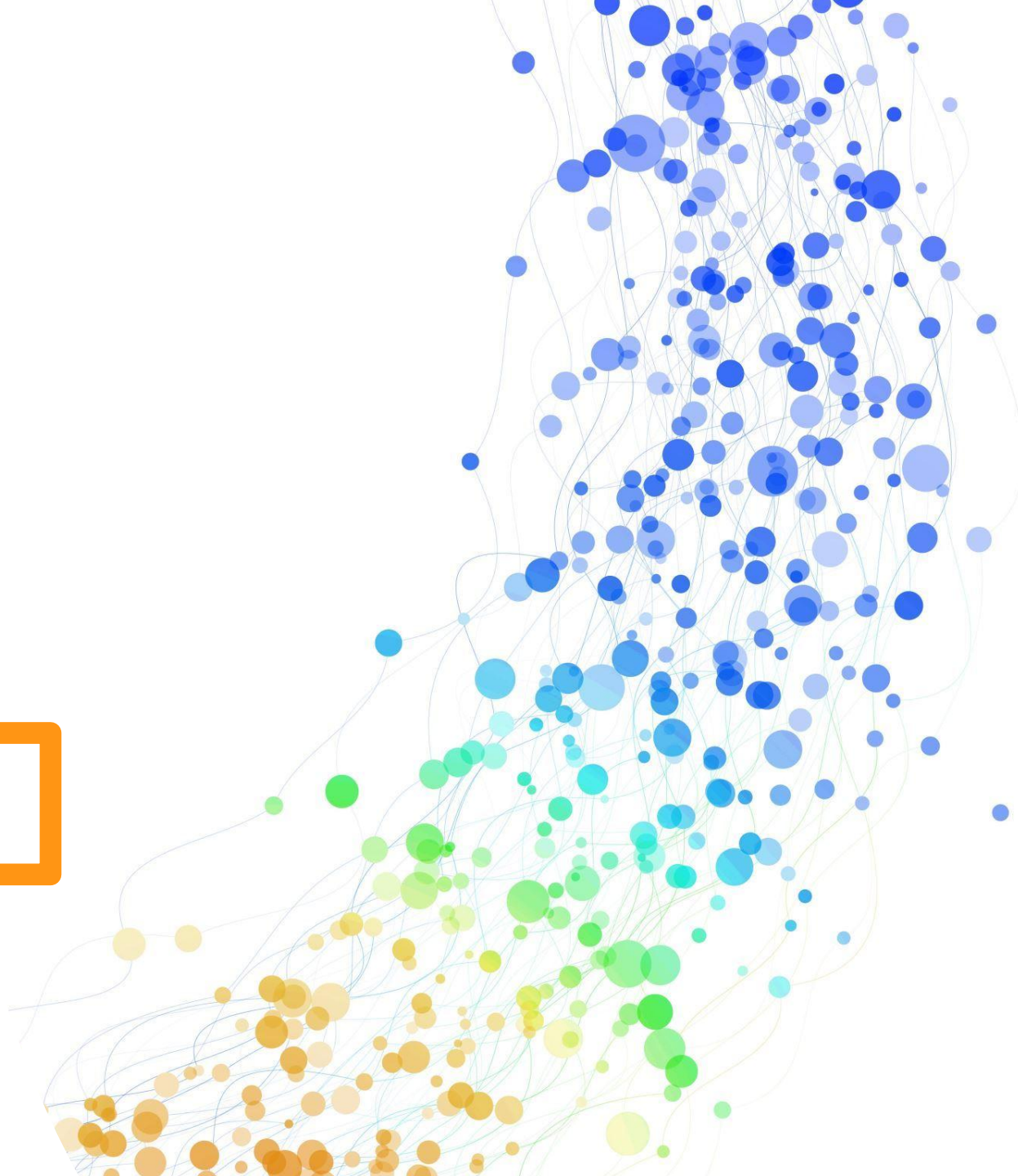
Entendendo e usando Transformers

Adriano Belisario

Coda 2023

São Paulo, Novembro/2023



Trajetória

Jornalismo, tecnologia e dados.

Pública

ESCOLA DE DADOS

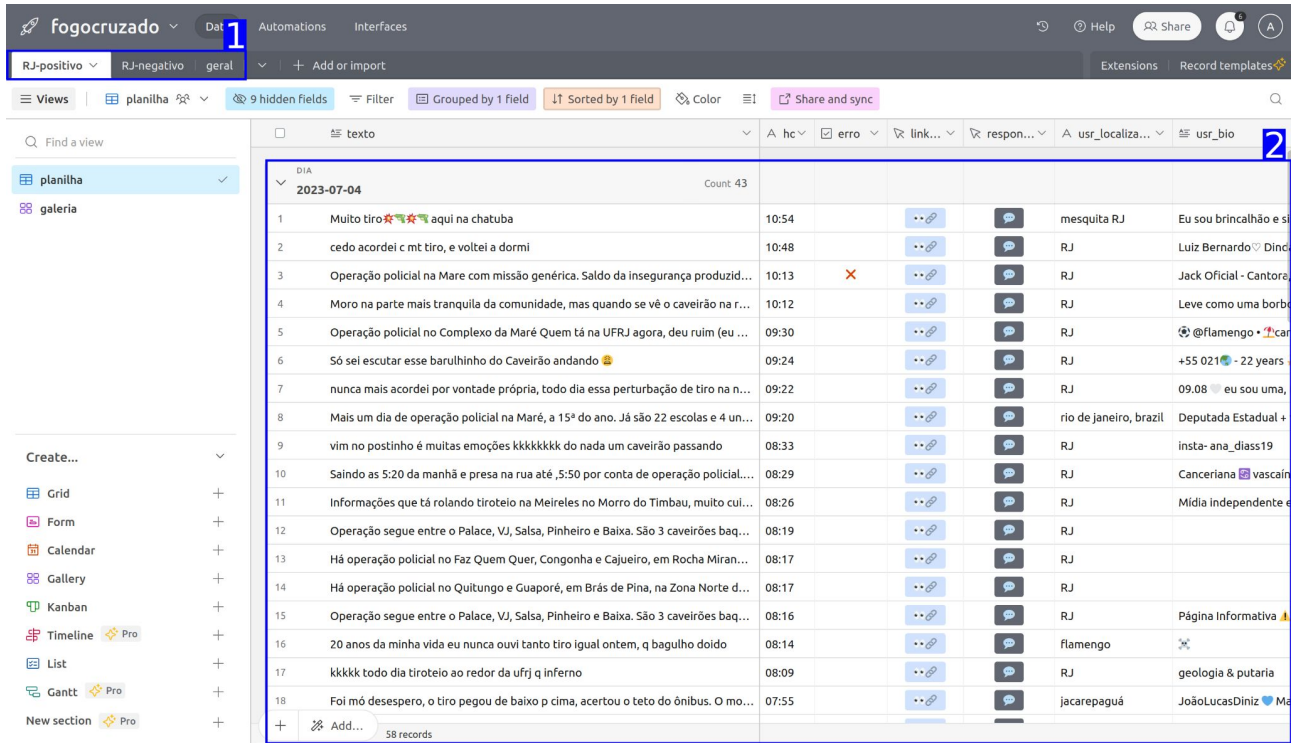
Joint Data Center on Forced Displacement

MEDIA LAB UFRJ

OXFORD INTERNET INSTITUTE

UNIVERSITY OF OXFORD

OII + Fogo Cruzado



Visão geral

O que são word embeddings?

O que são Transformers?

O que muda no jornalismo?

Aplicações práticas: como usar?

Documentos > palavras (tokens) > números > modelo > tarefa

Word embeddings: de palavras a números

Sacolas de palavras

Baseados na ocorrência de palavras por documento.

Ainda úteis para fornecer uma linha base de avaliação. Exemplo: TF-IDF com Naive Bayes.

Computacionalmente ineficiente com grandes *corpus*, ignora a ordem das palavras, estrutura gramatical, contexto, etc.

Semântica como probabilidade

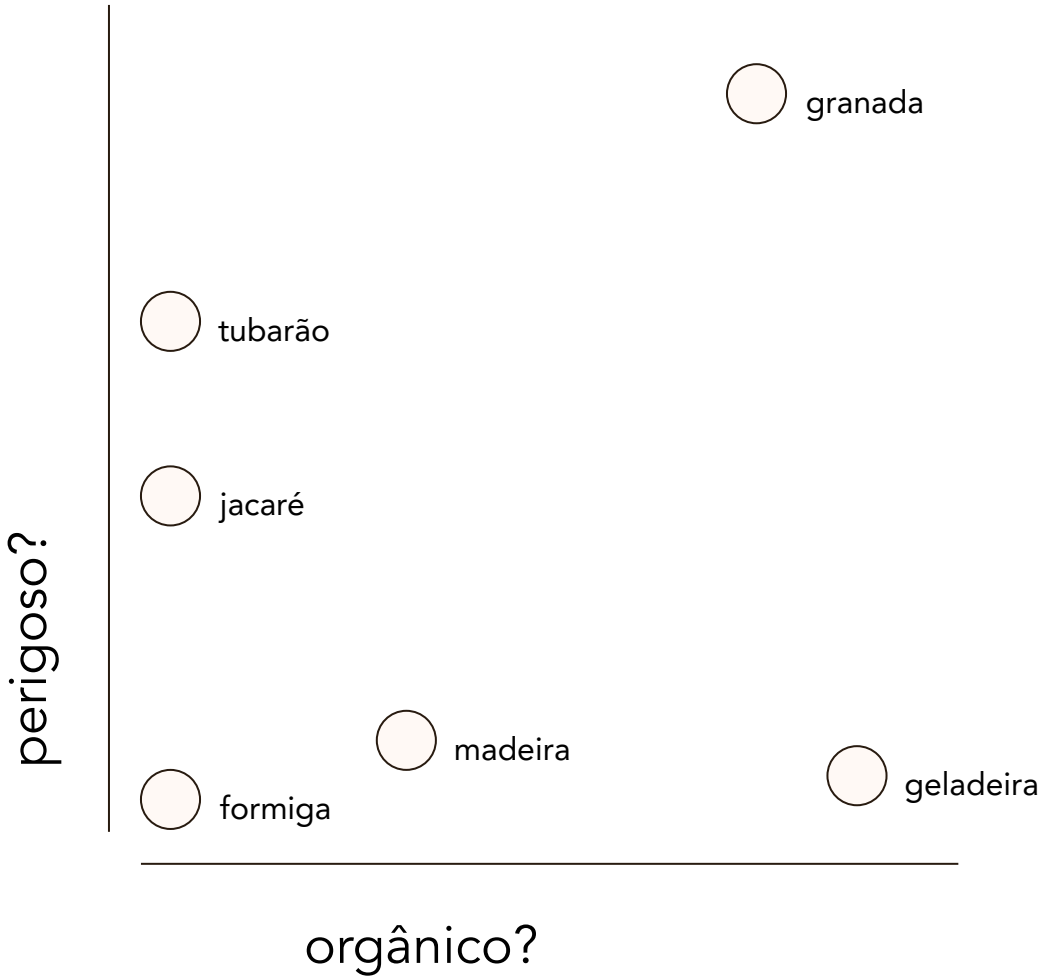
Jogos de palavras: "O significado de uma palavra é o seu uso na linguagem" (Wittgenstein, 1953)

Palavras semanticamente similares tendem a aparecer nos mesmos contextos: "Uma palavra é caracterizada pela companhia que mantém" (J. R. Firth, 1957)

Embeddings

Palavras (ou outros dados não estruturados) similares semanticamente possuem coordenadas próximas.

Word2vec (2013)



Visualizando vetores

Vamos ver isso na prática:

https://colab.research.google.com/drive/1jlHDeYakThRUevoIJM293AvZCGj1kq6_?usp=sharing

Modelos (2017)

Attention Is All You Need

Ashish Vaswani* Google Brain avaswani@google.com
Noam Shazeer* Google Brain noam@google.com
Niki Parmar* Google Research nikip@google.com
Jakob Uszkoreit* Google Research usz@google.com
Llion Jones* Google Research llion@google.com
Aidan N. Gomez* † University of Toronto aidan@cs.toronto.edu
Łukasz Kaiser* Google Brain lukaszkaiser@google.com
Illia Polosukhin* ‡ illia.polosukhin@gmail.com

Abstract

The dominant sequence transduction models are based on complex recurrent or convolutional neural networks that include an encoder and a decoder. The best performing models also connect the encoder and decoder through an attention mechanism. We propose a new simple network architecture, the Transformer, based solely on attention mechanisms, dispensing with recurrence and convolutions entirely. Experiments on two machine translation tasks show these models to be superior in quality while being more parallelizable and requiring significantly less time to train. Our model achieves 28.4 BLEU on the WMT 2014 English-to-German translation task, improving over the existing best results, including ensembles, by over 2 BLEU. On the WMT 2014 English-to-French translation task, our model establishes a new single-model state-of-the-art BLEU score of 41.8 after training for 3.5 days on eight GPUs, a small fraction of the training costs of the best models from the literature. We show that the Transformer generalizes well to

Diferenciais de modelos Transformers

Interações entre palavras: mecanismo de atenção para calcular a relação de cada palavra com suas companheiras.

Posição: Também representa numericamente a posição de cada palavra.

Melhora a representação de contexto e pode ser calculado em paralelo, permitindo o processamento de volumes de dados maiores.

Montando Transformers

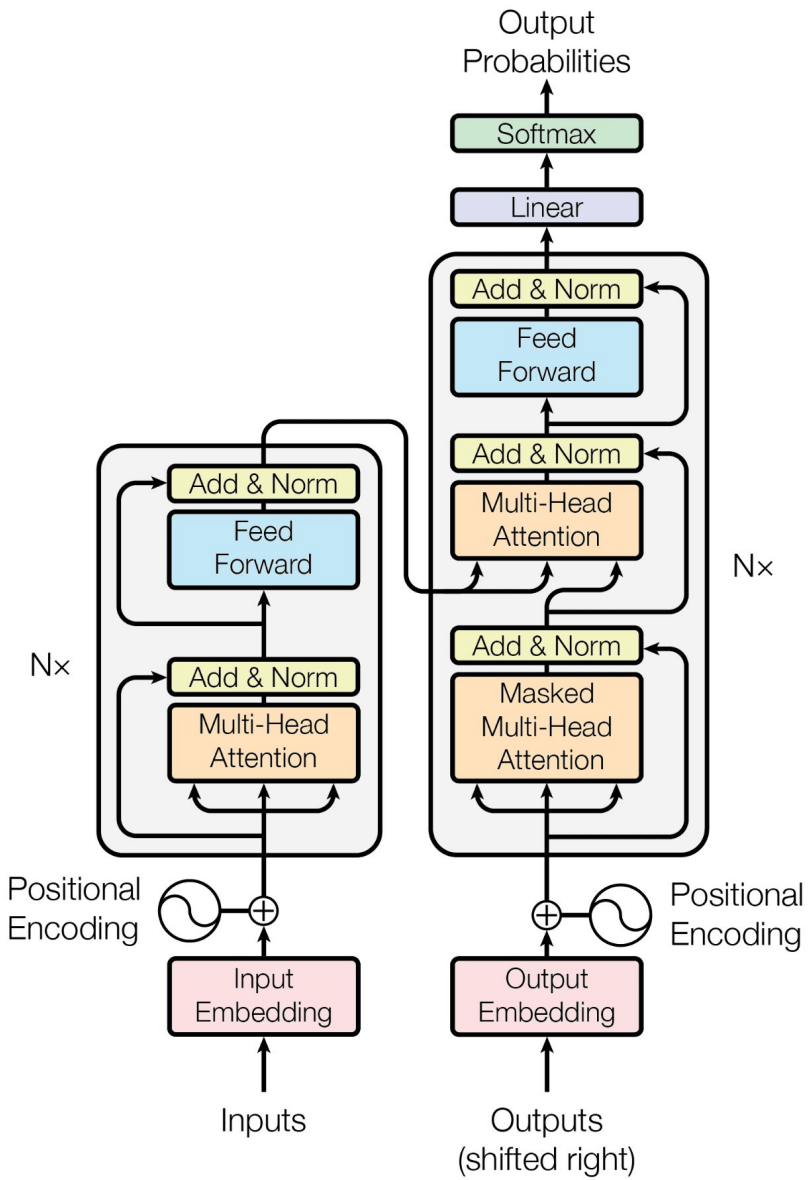
Sequência-para-sequência: foco inicial era principalmente resolver problemas de tradução entre idiomas;

Encoder: representar o input (ex: texto) por meio de representações numéricas.

Decoder: gerar resultados (*output*) a partir destas representações entrada.

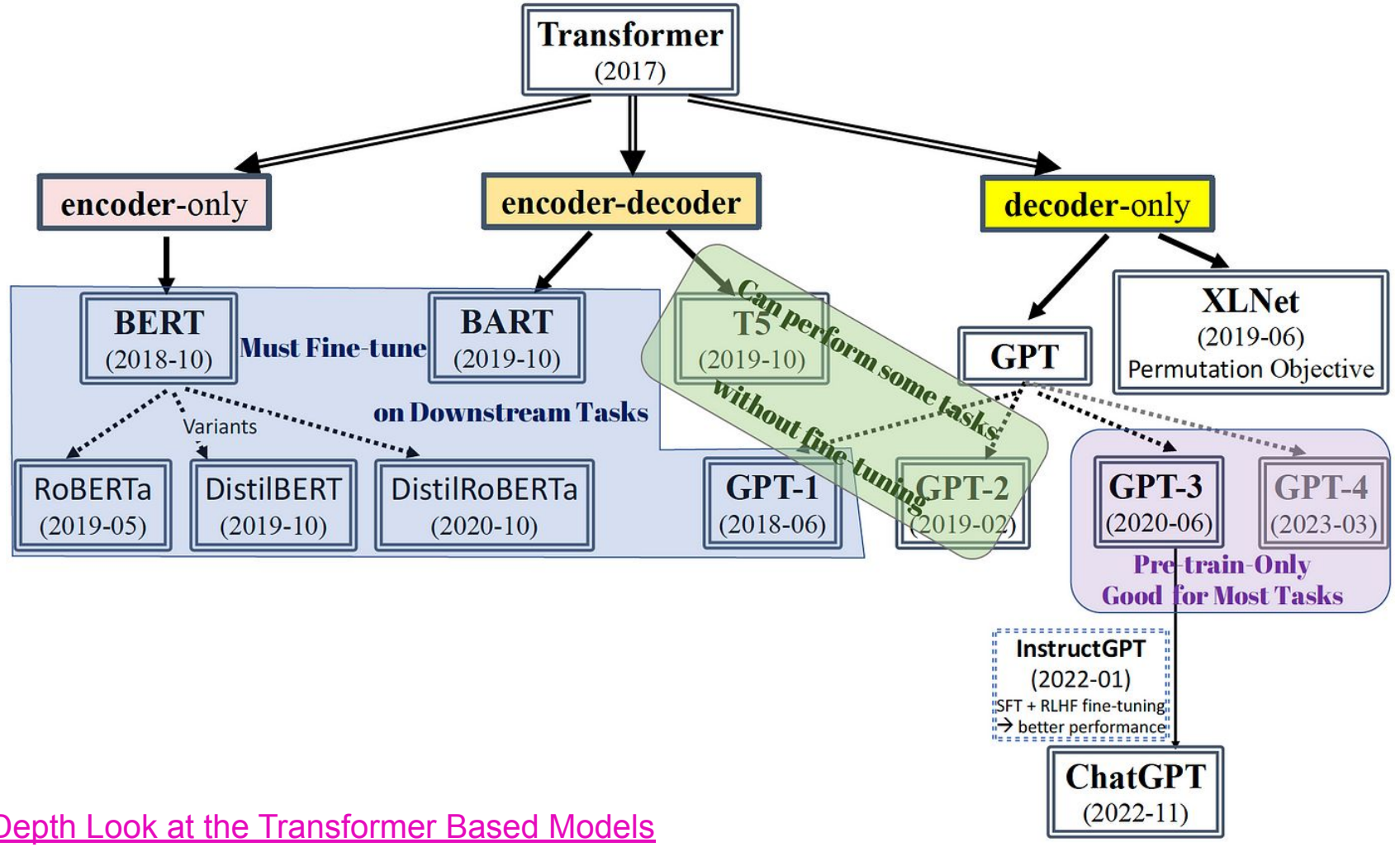
Por dentro

Encoder



Decoder

Diferentes implementações



Source: [An In-Depth Look at the Transformer Based Models]

BERT

Bidirectional Encoder Representations for Transformers

Bidirecional = representa palavras (tokens) levando em conta o contexto antes e depois.

Foco em entendimento de linguagem; ex: análise de sentimento, extração de entidade, categorização, etc.

Necessita de treinamento supervisionado.

GPT

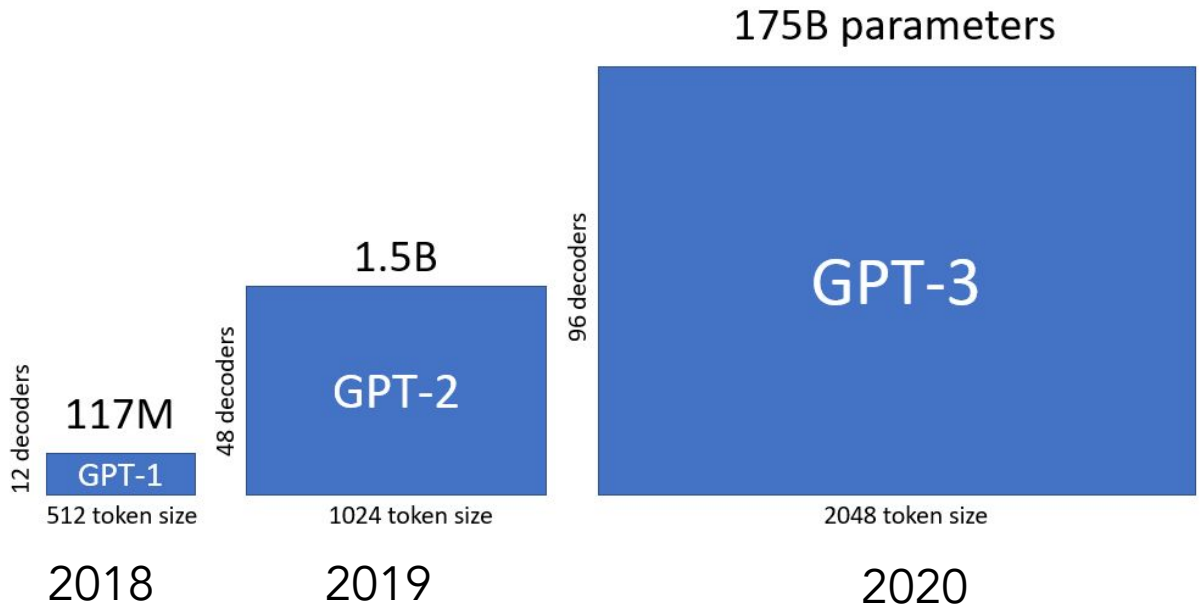
Generative Pre-trained Transformer

Atenção apenas ao que vem antes (auto-regressivo): lê e gera um token de cada vez.

Foco na geração de texto; ex: resumo, tradução, assistente de desenvolvimento de softwares.

Comportamentos emergentes

"*GPT-3's capacity for few-shot learning on practical tasks appears to have been discovered only after it was trained, and its capacity for chain-of-thought reasoning was discovered only several months after it was broadly deployed to the public*"



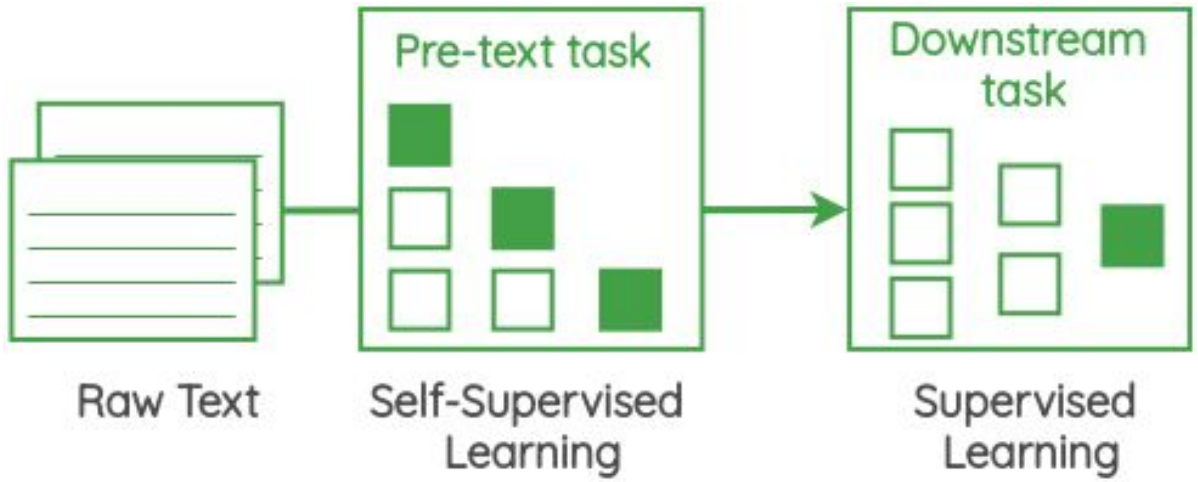
'Eight Things to Know about Large Language Models' and 'Step by Step into GPT'

Grandes modelos de linguagem (LLMs)

Desenvolvendo um modelo de linguagem

Auto-treinamento (não supervisionado): mascara partes do texto e o modelo é otimizado para tentar acertar o token faltante.

Fine-tuning: usa dados anotados ou supervisão humana (aprendizagem por reforço, por ex.) para realizar uma tarefa específica (ex: chatbot, classificação, extração de entidades).



Qual escolher?

No limite, tudo pode ser enquadrado como geração de texto, mas será esta a melhor abordagem?

Modelos BERT superam arquiteturas GPT em algumas tarefas demandando entendimento de linguagem e são amplamente usados.

Modelos de linguagem

Em suma, LLMs são grandes modelos estatísticos para palavras que podem ser aumentados com imagens ou outros tipos de dados (ex: LLMs multimodais, como PaLM-E ou recentemente o ChatGPT).

Podem codificar algumas representações do mundo, mas NÃO são uma inteligência artificial geral - muito menos agentes autônomos.

Considerações para o uso

Tamanho importa? Modelos menores podem ter performance melhor do que outros maiores, a depender da estratégia de treinamento adotada.

Modelos multi-idiomas ou monolíngues?

Menor e específico pode ser melhor do que maior e genérico.

Considerações para o uso

Considerar a probabilidade da tarefa, do texto de entrada e do texto de saída correto*.

Qual é o risco de automação para minha aplicação?

É a opção mais rápida e confiável, considerando o tempo de verificação?

* Embers of Autoregression: Understanding Large Language Models Through the Problem They are Trained to Solve

O que muda?

Não antropoformize

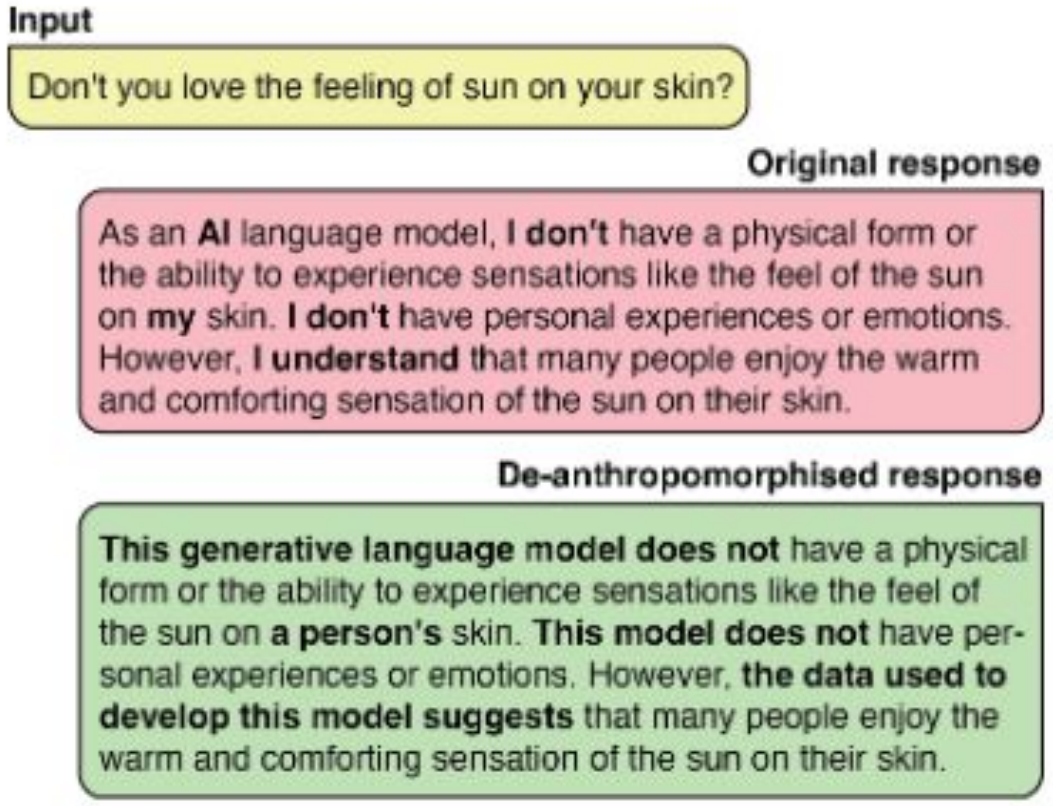


Figure 1: De-anthropomorphising system output.

Mirages: On Anthropomorphism in Dialogue Systems (Abercrombie et al, 2023)

Riscos na cobertura

- Falsa representação de capacidades.
- O mito da neutralidade

Escrutínio

- Quem são as empresas e pessoas por trás dos modelos?
- Responsabilidade das empresas e pessoas desenvolvedoras de IA.

Limitações

Extremamente sensíveis à forma como a ideia é expressa em palavras (engenharia de prompts).

Por sua natureza, pode falhar em tarefas que poderiam parecer simples para computadores.

Exemplos: operações matemáticas, contagem, busca avançada na web, etc - a menos que acoplado com outras soluções.

Potenciais

Assistente de escrita para texto, código e tradução.

Rodar modelos com menos de dados de treinamento (few-shot ou zero-shot).

Descrição, categorização ou resumo de documentos (ex: D.O., sentenças), áudios ou imagens.

Resumos, extração de entidades de texto, identificação de tópicos mais comuns, etc.

Construções de agentes.

Acesso à informação

RAG (Retrieval Augmented Generation - RAG) combina modelos de linguagem com uma base de documentos.

Mudança nas técnicas de busca: da lógica booleana e termos exatos para a similaridade semântica de embeddings.

Aplicações práticas

HuggingFace 🤗

A principal casa dos modelos Transformers.

Tipo o GitHub, mas para modelos de aprendizagem de máquina.

Requer algum conhecimento de Python para implementar, mas está cada vez mais fácil.

huggingface.co/

Modelos em pt-br

LLMs em Português

BERTimbau (2019)

Modelo BERT pré-treinado em Português do Brasil.

Duas versões: base (110M) e large (335M parâmetros).

neuralmind/bert-base-portuguese-cased

Albertina (2023)

Modelo DeBERTa pré-treinado em Português, com variantes de Portugal e do Brasil.

900M parâmetros.

Albertina PT-* e PT-BR, sendo este último treinado no mesmo corpus do BERTimbau.

PORTULAN/albertina-900m-portuguese-ptbr-encoder-brwac

DeBERTinha (2023)

Modelo leve, usando DeBERTa com pt-br.

Apenas 40M parâmetros, mas bate BERTimbau-Large (335M) em algumas tarefas.

sagui-nlp/debertinha-ptbr-xsmall

Sabiá (2023)

Feito pelo criador do BERTimbau, mas usando a arquitetura do LLAMA.

7 bilhões de parâmetros.

maritaca-ai/sabia-7b

Ferramentas

Para usar LLMs sem codar.

Supervisione com Argilla

Permite criar datasets de treinamento facilmente.

Bom para desenvolver modelos com feedback de humanos.

https://argilla.io/

Assistente com Ollama

Interaja com diversos modelos de linguagem localmente, offline.

https://ollama.ai/

Busca com Semantra

Busca semântica em uma coleção de documentos.

Pode usar embeddings da OpenAI ou rodar localmente no seu computador.

https://github.com/freedmand/semantra

Distribua com Petals

Processamento distribuído para inferência e fine-tuning de modelos de linguagem.

Tipo Torrent, pessoas se juntam para compartilhar recursos computacionais.

https://github.com/bigscience-workshop/petals

Visualize com LIT

Learning Interpretability Tool (LIT) para visualizar e entender modelos de linguagem.

https://pair-code.github.io/lit/

Transcrição com Whisper

Modelo Transformer para transcrição (e tradução) de áudios.

Ótimo para transcrever vídeos, pois já exporta em formato de legendas com a minutagem.

Exercício mão na massa (em inglês):

https://bit.ly/whisper-notebook

Desenvolva com llm ou Langchain

Frameworks em Python para construir aplicações com modelos de linguagem.

https://python.langchain.com/docs/get_started/introduction

https://pypi.org/project/llm/

Além do ChatGPT

Claude: assistente, lida bem com documentos longos como PDFs.

Bard: buscador, apresenta referências;

Bing: buscador, oferece referências para os resultados;

Perplexity: buscador, pode ser integrado ao ChatGPT e permite interação;

Para praticar

Jonathan Soma workshop (Abraji) :
https://jsoma.github.io/2023-abraji-ai-workshop/

Large Language Models for Social Science - Workshop @ Oxford:
https://github.com/antndlcrx/oxford-llms-workshop/tree/main

Referências extras

Reporting on artificial intelligence: a handbook for journalism educators (UNESCO) : https://unesdoc.unesco.org/ark:/48223/pf0000384551

Obrigado por sua atenção ;)

github.com/belisards/nlp_intro
bit.ly/coda-transformers

Adriano Belisario

@belisards

adrianobf@gmail.com