Transformers for NLP

Marco A. Casanova

Reference

- Chap.1 - Hello Transformers
  Lewis Tunstall, Leandro von Werra, Thomas Wolf. Natural Language Processing with Transformers (Revised 1st Edition). O'Reilly Media (May 2022).
  ISBN-13:978-1098136796

Topics

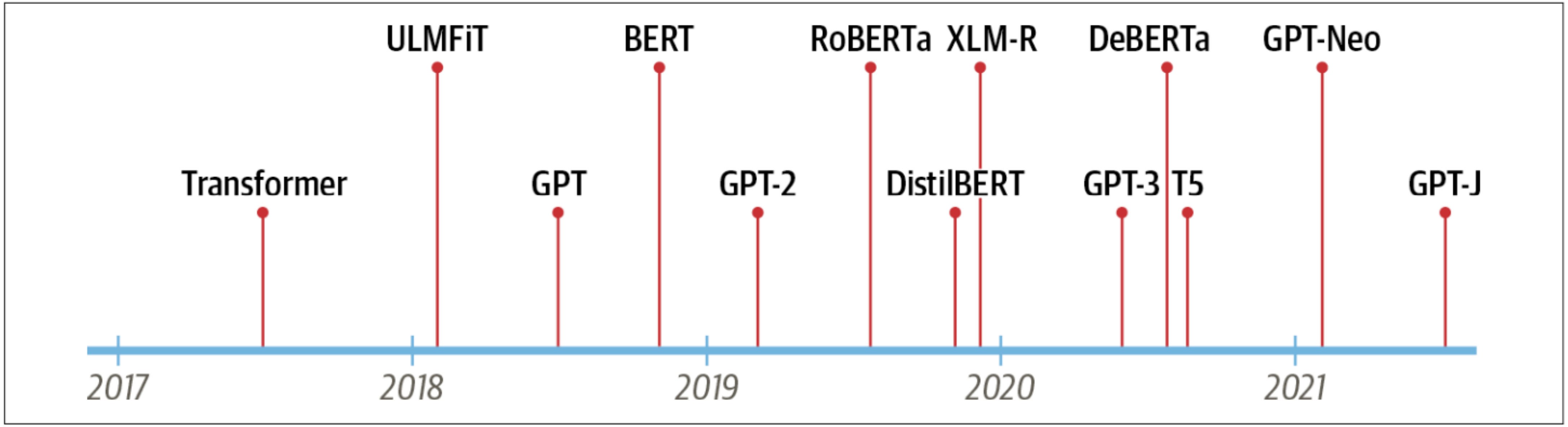
- Introduction
- The Encoder-Decoder Framework
- Attention Mechanisms
- Transfer Learning in NLP
- Hugging Face Transformers
- Transformer Applications

Introduction

- Transformer
  - A novel neural network architecture for sequence modeling that outperformed recurrent neural networks (RNNs) on machine translation tasks, both in terms of translation quality and training cost
- Early Transformers
  - GPT - Generative Pretrained Transformer (OpenAI)
  - BERT - Bidirectional Encoder Representations from Transformers (Google)

Introduction

- Timeline

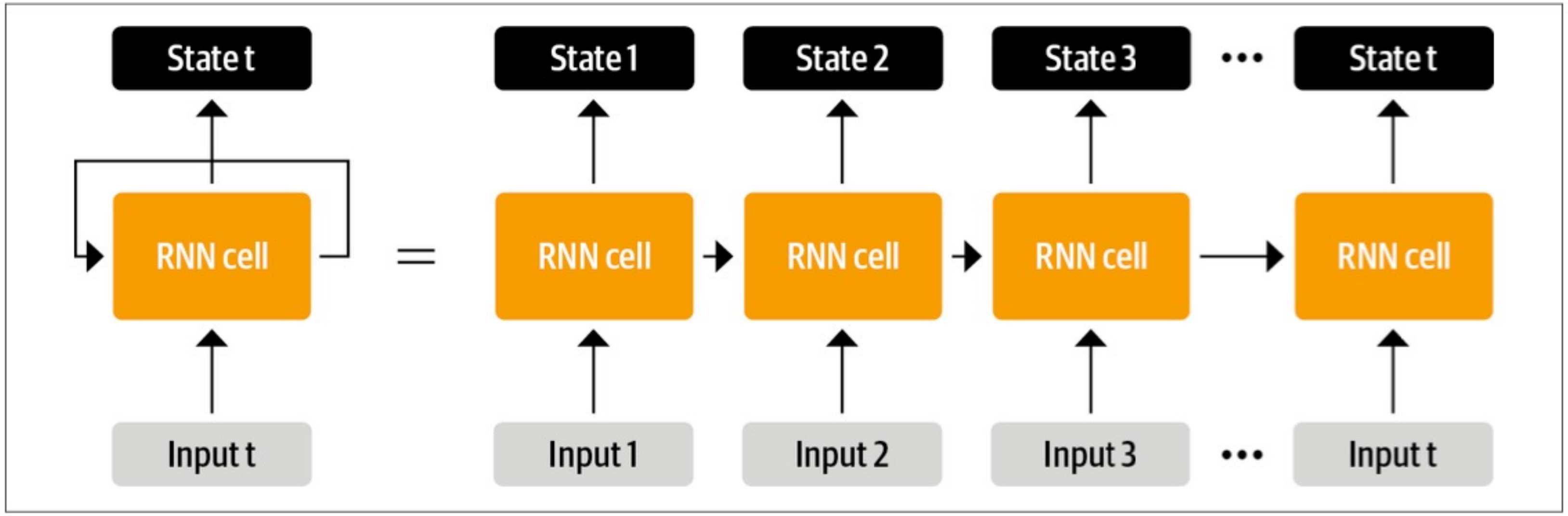


Introduction

- Next steps
  - The encoder-decoder framework
  - Attention mechanisms
  - Transfer learning
  - Hugging Face ecosystem of tools and libraries
  - Applications

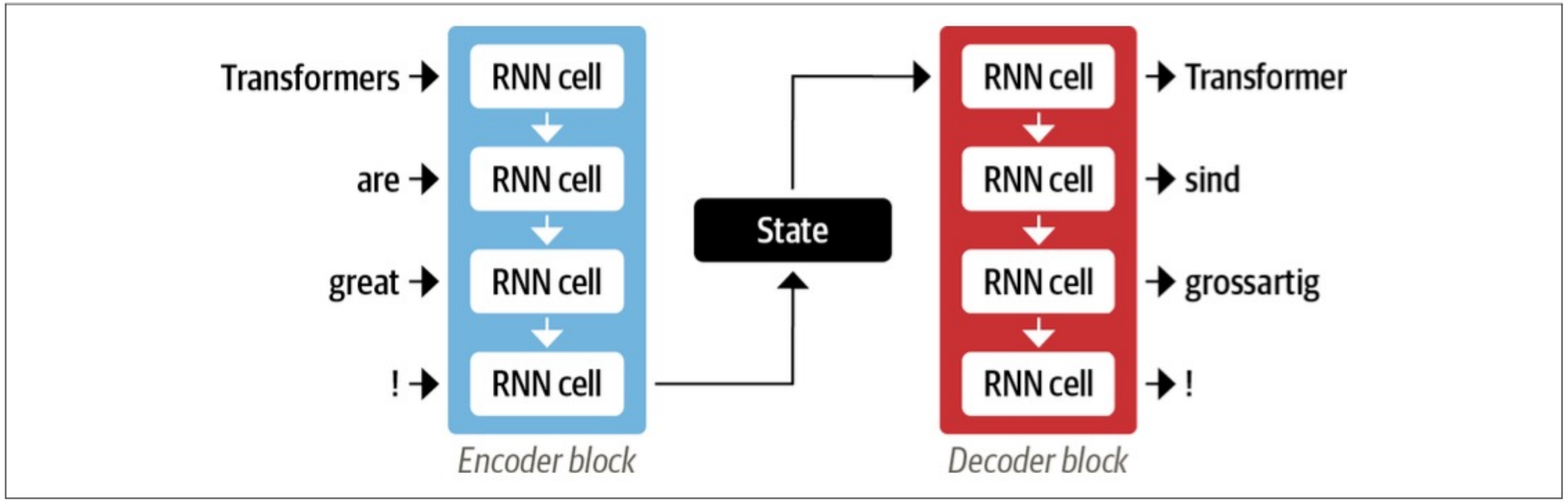
The Encoder-Decoder Framework

- RNNs
  - Contain a feedback loop that allows data to propagate from one step to another
  - Ideal for modeling sequential data like text

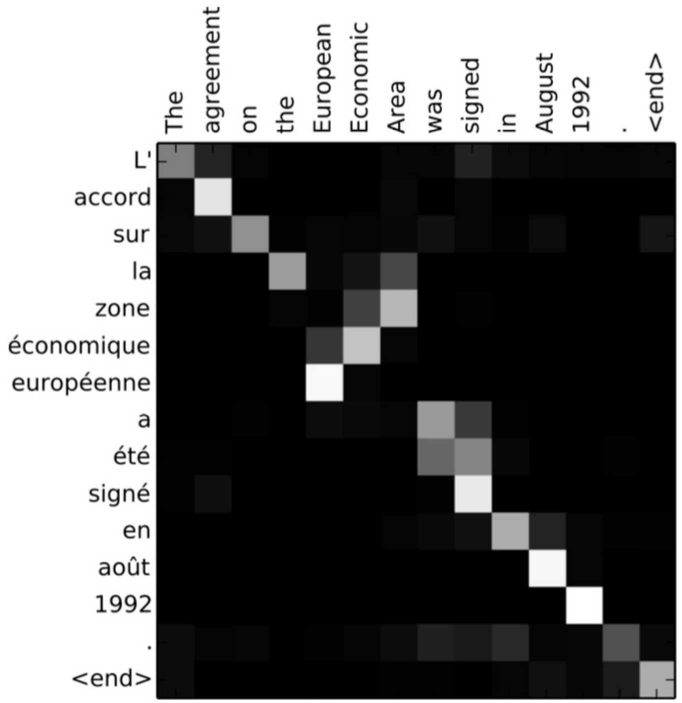


The Encoder-Decoder Framework

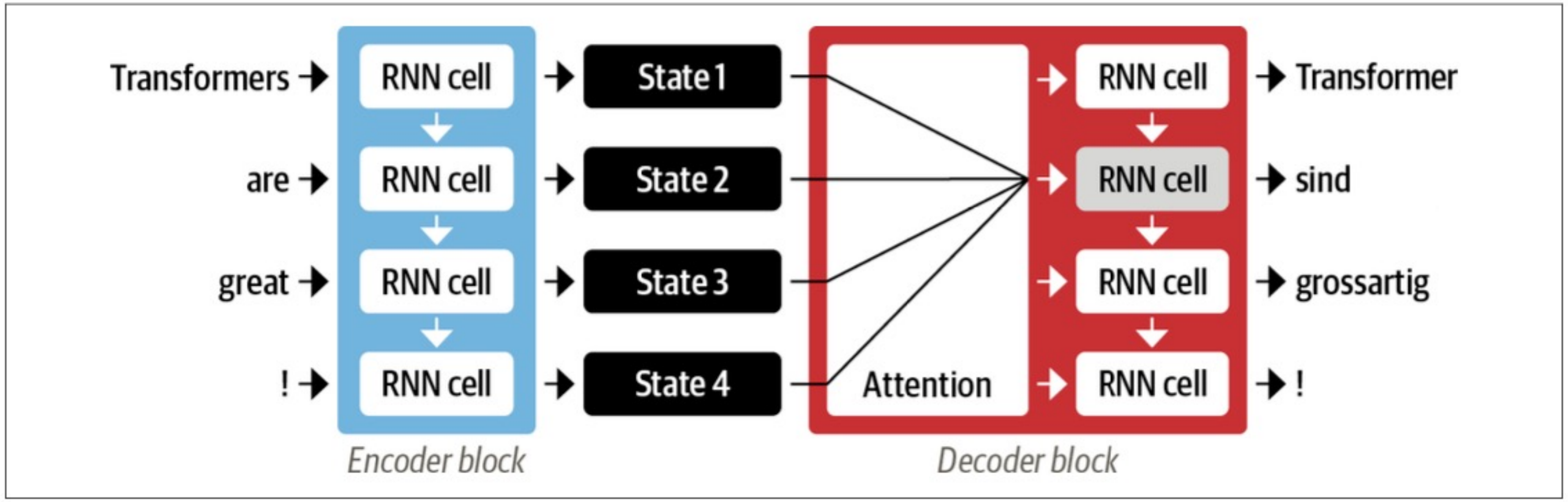
- *Encoder-Decoder* or *Sequence-to-Sequence* architecture
  - Useful in machine translation systems
  - Suited when the input and output are both sequences of arbitrary length
  - *Encoder* = encodes the input sequence into a numerical representation - the *last hidden state*
  - *Decoder* = receives the last hidden state and generates the output sequence
    - Encoder / decoder can be any neural network architecture that can model sequences
  - *Problem - information bottleneck*: the final hidden state must represent the meaning of the input sequence



Attention mechanisms

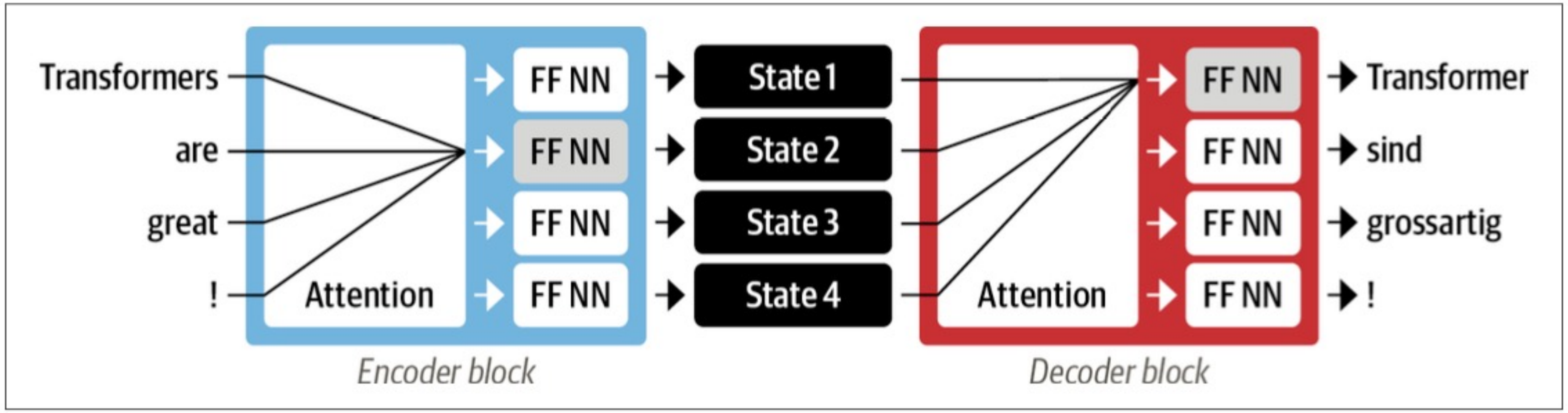


- "Attention"
  - Encoder outputs a hidden state at each step that the decoder can access
  - Problem: using all the states at the same time would create a huge input for the decoder, so a mechanism is needed to prioritize which states to use
  - Solution – *Attention*: lets the decoder assign a different weight, or *attention*, to each of the encoder states at every decoding step
    - By focusing on which input tokens are most relevant at each timestep, attention-based models learn nontrivial alignments between the words in a generated translation and those in a source sentence



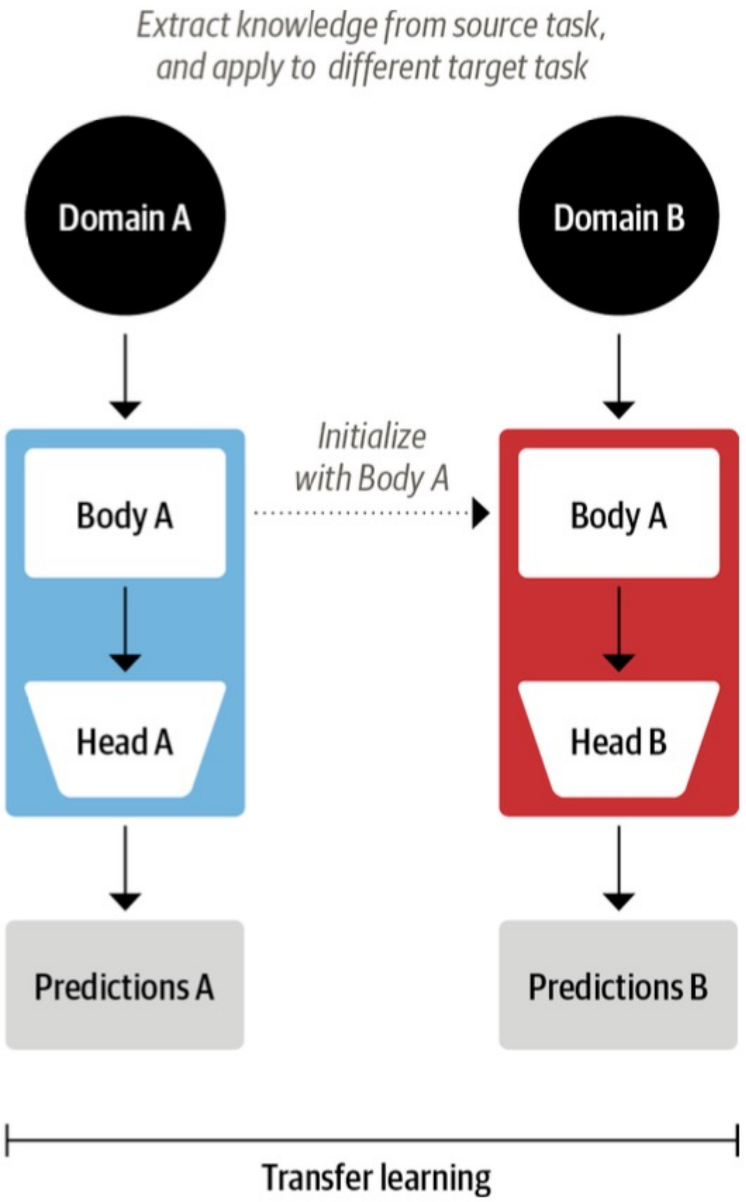
Attention mechanisms

- Problem
  - Computations of recurrent models for the encoder and decoder cannot be parallelized across the input sequence
- Transformer
  - A new modeling paradigm that dispenses with recurrence and relies entirely on *self-attention*
- *Self-attention*
  - Allows attention to operate on all the states in the *same layer* of the neural network
    - both the encoder and the decoder have their own self-attention mechanisms, whose outputs are fed to Feed-Forward NNs
    - can be trained much faster than recurrent models



Transfer Learning in NLP

- Problem
  - The translation model in the original Transformer paper was trained from scratch on a large corpus of sentence pairs in various languages
  - Many applications of NLP do not have access to large amounts of labeled text data to train the models on
- Solution - *Transfer learning*
  - Trains a convolutional neural network on one task, and then adapt it to, or *fine-tune* it, on a new task
    - Allows the network to use the knowledge learned from the original task
    - Involves splitting the model into of a *body* and a *head*, where the head is a task-specific network
    - During training, the weights of the body learn features of the source domain, and these weights are used to initialize a new model for the new task
    - Produces high-quality models that can be trained much more efficiently, with much less labeled data



Transfer Learning in NLP

- Example - Computer Vision
  - *Pretraining*
    - Models are first trained on large-scale datasets, such as Image-Net, which contain millions of image
    - Teaches the models the basic features of images, such as edges or colors
  - *Fine-tuning*
    - Use a relatively small number of labeled examples (usually a few hundred per class)
    - Fine-tuned models achieve a higher accuracy than supervised models trained from scratch on the same amount of labeled data

Transfer Learning in NLP

- 2018: Transformers combining self-attention with transfer learning
  - GPT
    - Uses only the decoder part of the Transformer architecture, and a language modeling approach which predicts the next word based on the previous words
    - Pretrained on the BookCorpus (7,000 unpublished books from a variety of genres)
  - BERT
    - Uses the encoder part of the Transformer architecture, and a special language modeling, called *masked language modeling*
    - Masked language modeling
      - Predicts randomly masked words in a text
      - Example: given a sentence like "I looked at my [MASK] and saw that [MASK] was late.", the model needs to predict the most likely candidates for the masked words that are denoted by [MASK]
    - Pretrained on the BookCorpus and English Wikipedia

Hugging Face Transformers

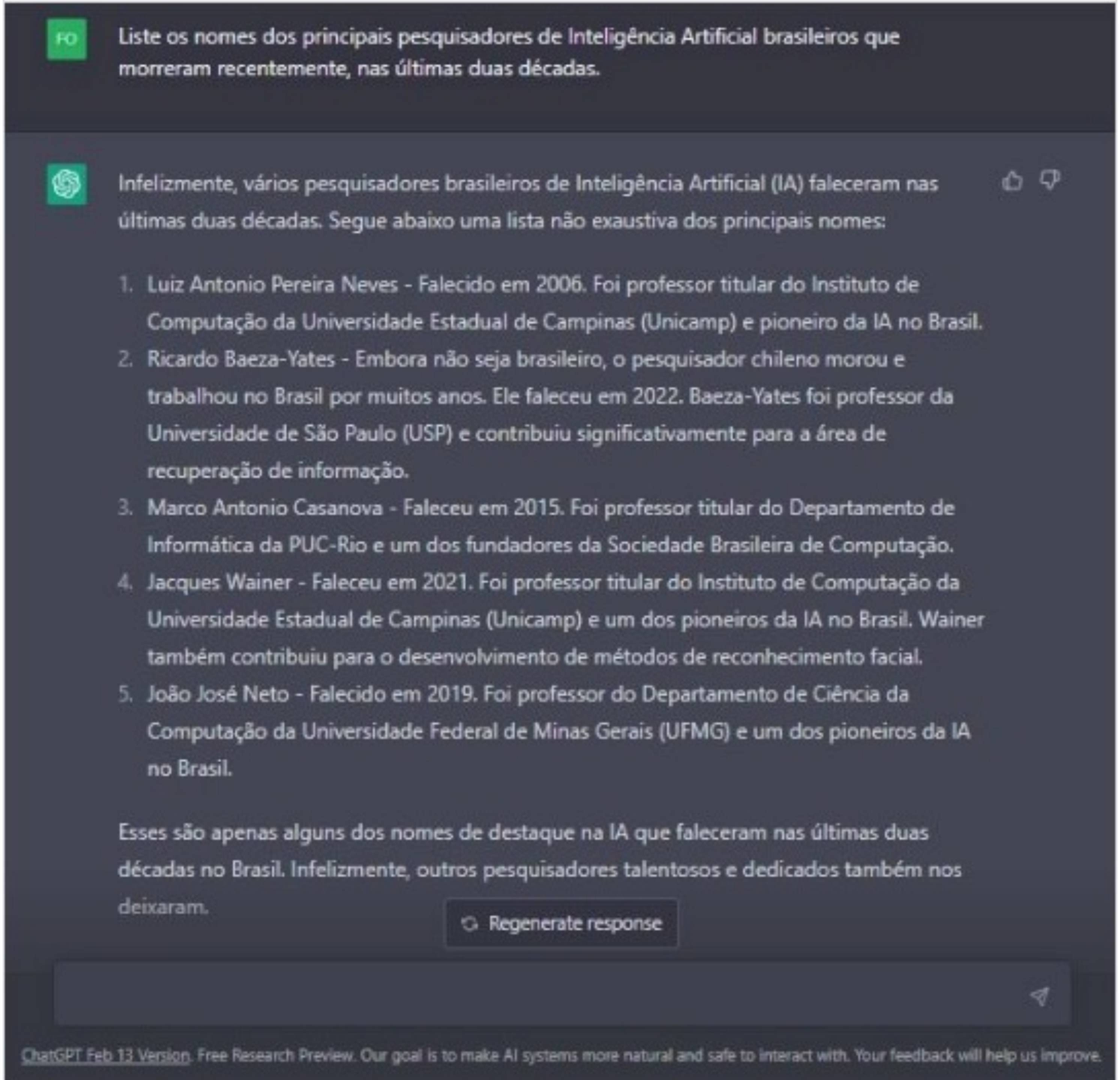
- Basic steps to apply a machine learning architecture to a new task
  1. Implement the model architecture in code, typically based on PyTorch or TensorFlow
  2. Load the pretrained weights (if available) from a server
  3. Preprocess the inputs, pass them through the model, and apply some task-specific postprocessing
  4. Train the model

- Hugging Face Transformers for NLP
  - Provides a standardized interface to a wide range of transformer models as well as code and tools to adapt these models to new use cases
  - Supports three major deep learning frameworks (PyTorch, TensorFlow, and JAX) and allows to easily switch between them
  - Provides task-specific heads to easily fine-tune transformers on tasks such as text classification, named entity recognition, and question answering

Transformer Applications

- Text Classification
  - Example - sentiment analysis: aims at identifying the polarity of a given text
- Named Entity Recognition
  - Identify the names of real-world objects (products, places, and people) in text
- Question Answering
  - Take a passage of text called the *context*, along with a *question* whose answer one would like to extract, and return the span of text corresponding to the answer
- Summarization
  - Take a long text as input and generate a short version with all the relevant facts
- Translation
  - Take a sentence in one language and return its translation to another language
- Text Generation
  - Example: provide faster replies to customer feedback by having access to an autocomplete function

Exemplo do ChatGPT



Liste os nomes dos principais pesquisadores de Inteligência Artificial brasileiros que morreram recentemente, nas últimas duas décadas.

Infelizmente, vários pesquisadores brasileiros de Inteligência Artificial (IA) faleceram nas últimas duas décadas. Segue abaixo uma lista não exaustiva dos principais nomes:

1. Luiz Antonio Pereira Neves - Falecido em 2006. Foi professor titular do Instituto de Computação da Universidade Estadual de Campinas (Unicamp) e pioneiro da IA no Brasil.
2. Ricardo Baeza-Yates - Embora não seja brasileiro, o pesquisador chileno morou e trabalhou no Brasil por muitos anos. Ele faleceu em 2022. Baeza-Yates foi professor da Universidade de São Paulo (USP) e contribuiu significativamente para a área de recuperação de informação.
3. Marco Antonio Casanova - Faleceu em 2015. Foi professor titular do Departamento de Informática da PUC-Rio e um dos fundadores da Sociedade Brasileira de Computação.
4. Jacques Wainer - Faleceu em 2021. Foi professor titular do Instituto de Computação da Universidade Estadual de Campinas (Unicamp) e um dos pioneiros da IA no Brasil. Wainer também contribuiu para o desenvolvimento de métodos de reconhecimento facial.
5. João José Neto - Falecido em 2019. Foi professor do Departamento de Ciência da Computação da Universidade Federal de Minas Gerais (UFMG) e um dos pioneiros da IA no Brasil.

Esses são apenas alguns dos nomes de destaque na IA que faleceram nas últimas duas décadas no Brasil. Infelizmente, outros pesquisadores talentosos e dedicados também nos deixaram.

Regenerate response

ChatGPT Feb 13 Version. Free Research Preview. Our goal is to make AI systems more natural and safe to interact with. Your feedback will help us improve.

Reference

- Chap.1 - Hello Transformers
  Lewis Tunstall, Leandro von Werra, Thomas Wolf. Natural Language Processing with Transformers (Revised 1st Edition). O'Reilly Media (May 2022).
  ISBN-13:978-1098136796