dataiku

Introduction to LLMs

DEFINITIONS

Generative AI AI systems that can produce realistic content *(text, image, etc.)*

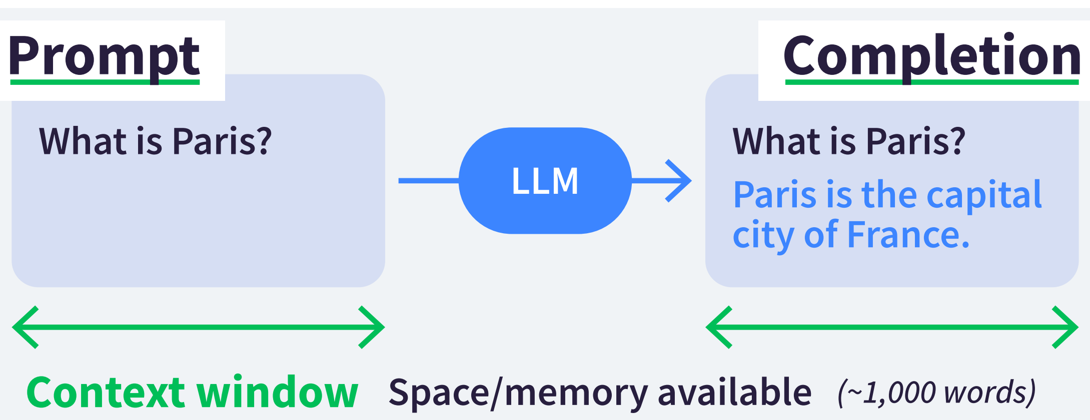
Large Language Models (LLMs)
Large neural networks trained at internet scale to estimate the probability of sequences of words

Ex: GPT, FLAN-T5, LLaMA, PaLM, BLOOM (transformers with billions of parameters)

Abilities (and **computing resources** needed) tend to rise with the number of parameters

USE CASES

- Standard NLP tasks *(classification, summarization, etc.)*
- Content generation
- Reasoning *(Q&A, planning, coding, etc.)*



In-context learning Specifying the task to perform directly in the prompt

Zero-Shot

Label this review:
Amazing product!
Sentiment:

One-Shot

Label this review:
Very high quality!
Sentiment: **Positive**

Label this review:
Amazing product!
Sentiment:

Few-Shot

Label this review:
Very high quality!
Sentiment: **Positive**

Label this review:
I don't really like it
Sentiment: **Negative**

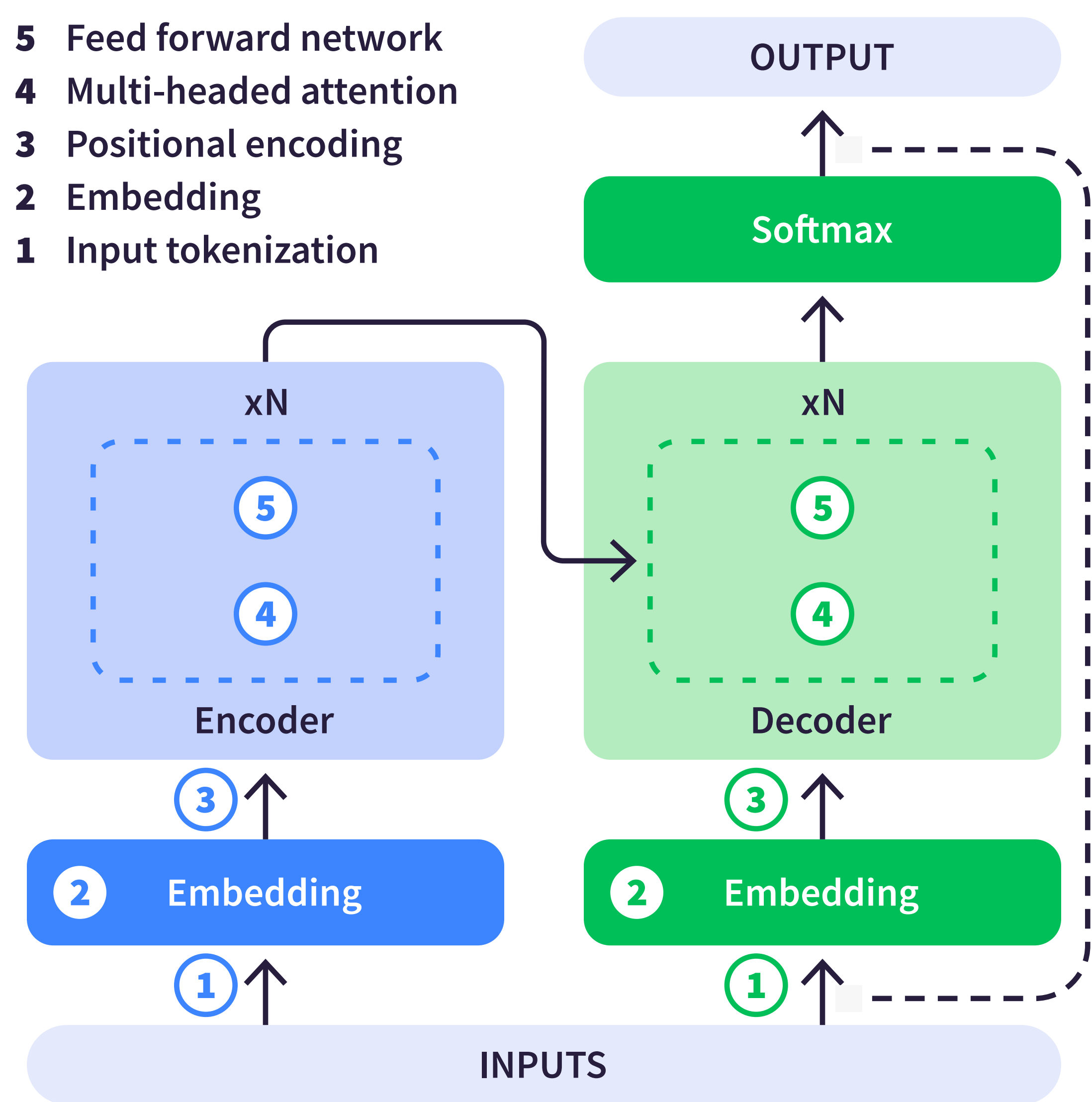
Label this review:
Amazing product!
Sentiment:

Include only a few examples (typically five). Consider fine-tuning if many examples are needed.

TRANSFORMERS

- Can scale efficiently to use multi-core GPUs
- Can process input data in parallel
- Pay attention to all other words when processing a word

Transformers' strength lies in understanding the **context** and **relevance** of all words in a sentence



Token Word or sub-word
The basic unit processed by transformers

Encoder Processes input sequence to generate a vector representation (or embedding) for each token

Decoder Processes input tokens to produce new tokens

Embedding layer Maps each token to a trainable vector

Positional encoding vector
Added to the token embedding vector to keep track of the token's position

Self-Attention Computes the importance of each word in the input sequence to all other words in the sequence

TYPES OF LLMS

Encoder only = **Autoencoding model**

Ex: BERT, RoBERTa
These are not generative models.

I like to MASK on weekends

Bidirectional context

PRE-TRAINING OBJECTIVE To predict **tokens masked** in a sentence (= **Masked Language Modeling**)

OUTPUT Encoded representation of the text

USE CASE(S) Sentence classification *(e.g., NER)*

Decoder only = **Autoregressive model**

Ex: GPT, BLOOM

I like to ?

Unidirectional context

PRE-TRAINING OBJECTIVE To predict the **next token** based on the previous sequence of tokens (= **Causal Language Modeling**)

OUTPUT Next token

USE CASES Text generation

Encoder-Decoder = **Seq-to-seq model**

Ex: T5, BART

I MASK MASK books on weekends

I X books on weekends

Sentinel token
Consecutive span of corrupted tokens added to the vocabulary

PRE-TRAINING OBJECTIVE Vary from model to model *(e.g., Span corruption like T5)*

OUTPUT Sentinel token + predicted tokens

USE CASES Translation, Q&A, summarization

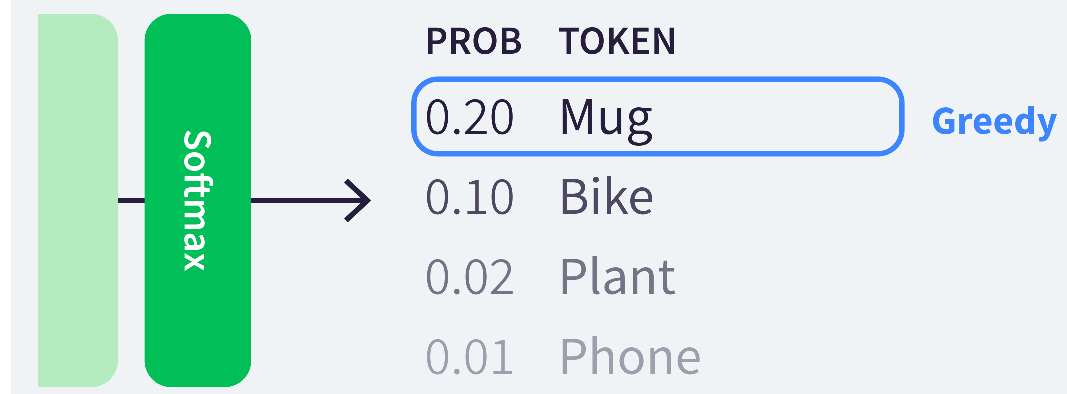
CONFIGURATION SETTINGS

Parameters to set at **inference time**

Max new tokens Maximum number of tokens generated during completion

Decoding strategy

1 Greedy Decoding The word/token with the highest probability is selected from the final probability distribution (prone to repetition)



2 Random Sampling The model chooses an output word at random using the probability distribution to weigh the selection (could be too creative)

TECHNIQUES TO CONTROL RANDOM SAMPLING

- **Top K** The next token is drawn from the **k** tokens with the highest probabilities
- **Top P** The next token is drawn from the tokens with the highest probabilities, whose combined probabilities exceed **p**

Temperature Influence the shape of the probability distribution through a scaling factor in the softmax layer