ORTA DOĞU TEKNİK ÜNİVERSİTESİ
MIDDLE EAST TECHNICAL UNIVERSITY

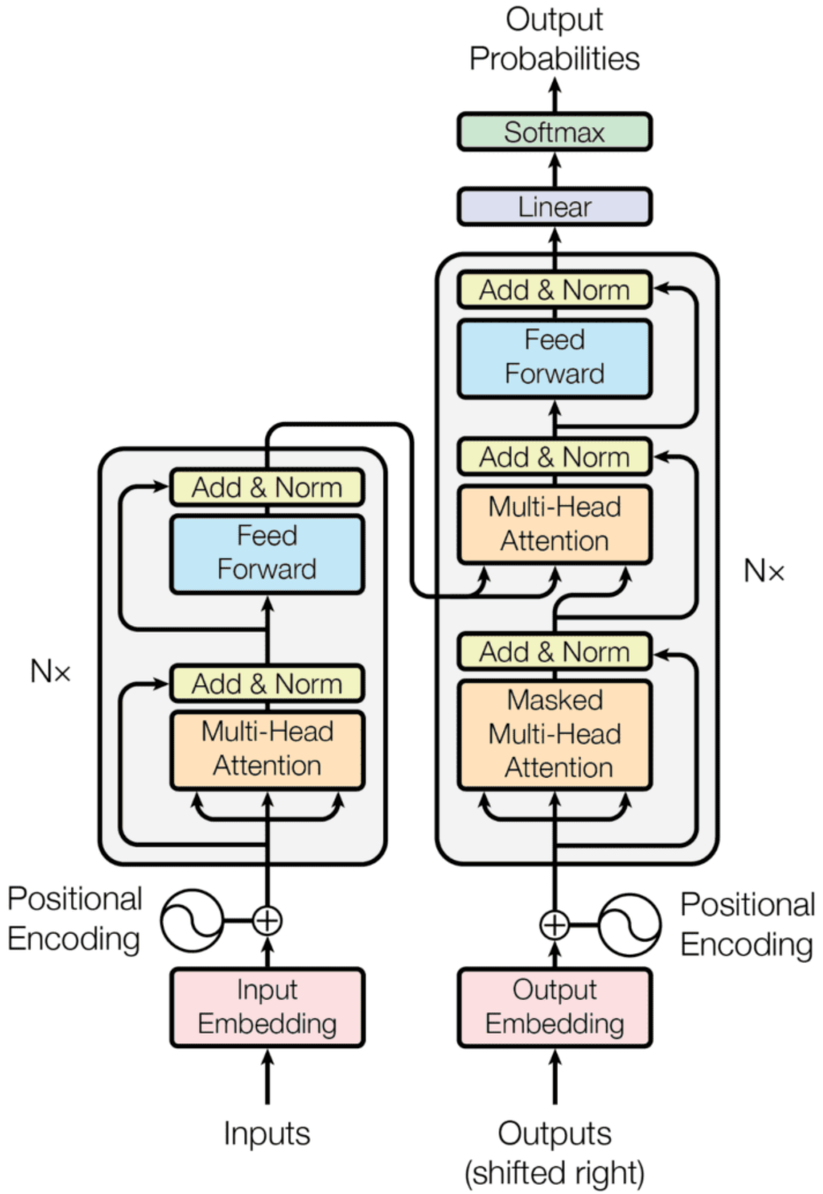
CENG 463: Introduction to Natural Language Processing
Large Language Models (Generative Models)

Asst. Prof. Çağrı Toraman
Computer Engineering Department
ctoraman@ceng.metu.edu.tr

16.12.2025

Transformer

ENCODER

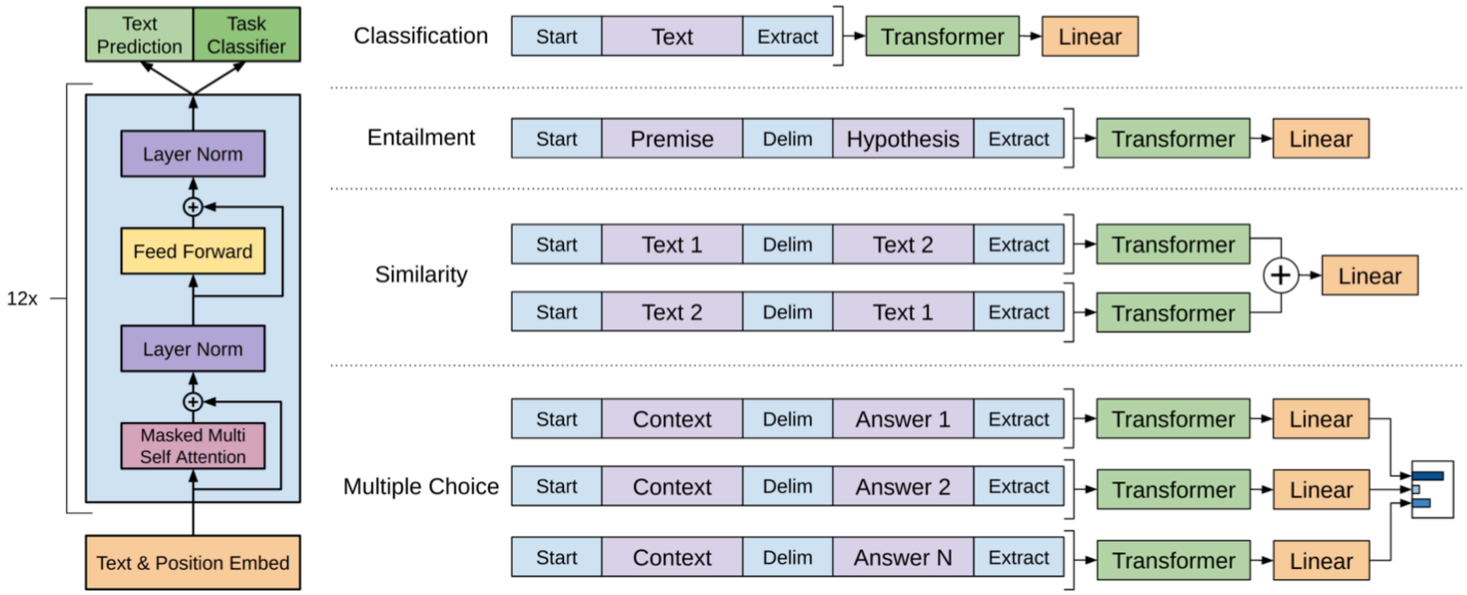


DECODER

A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. Gomez, L. Kaiser, I. Polosukhin, Attention is all you need, NeurIPS 2017

Decoder-based Models (GPT)

Generative Pre-Training (Radford et al, 2018)



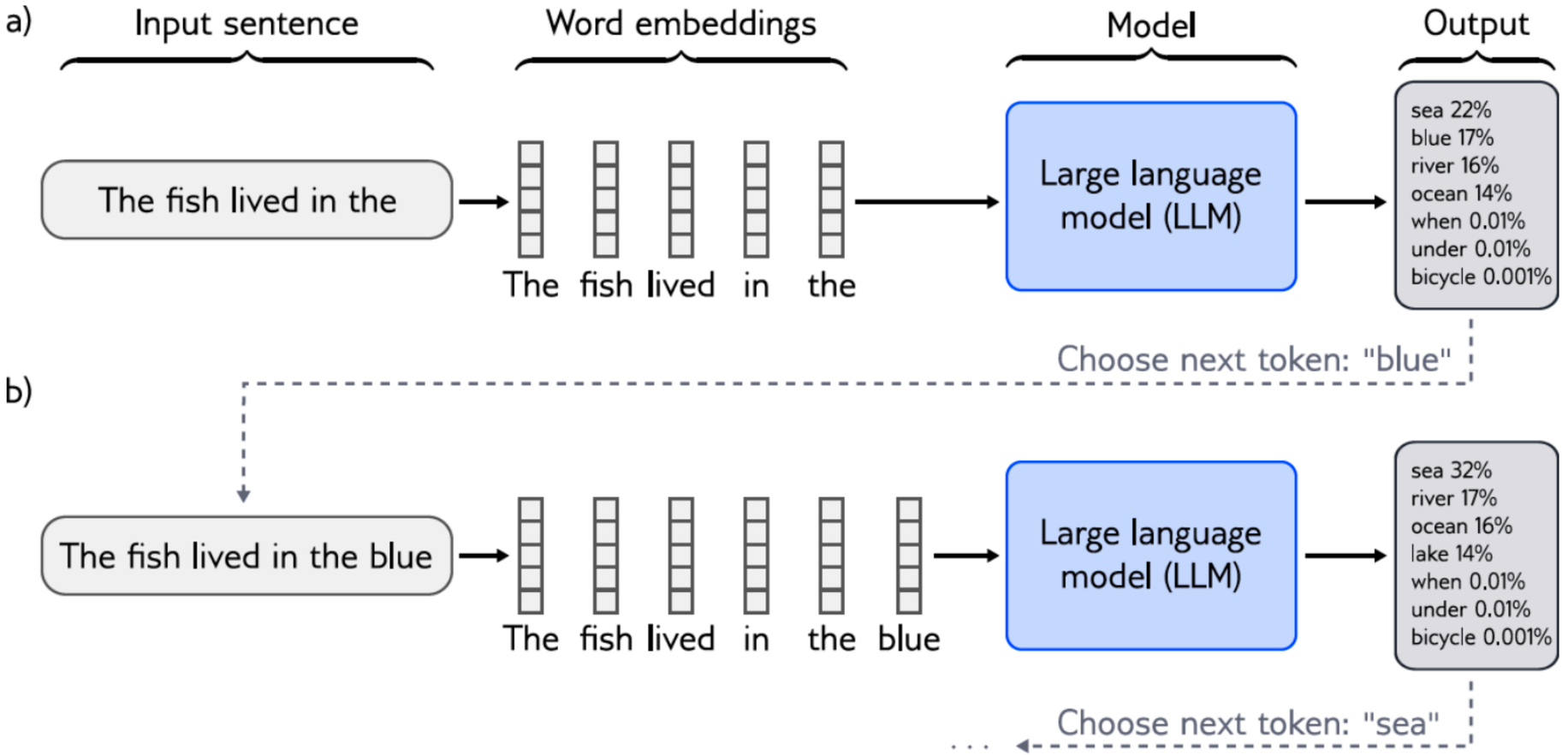
Auto-regressive 12-layer transformer decoder

BPE tokenization ($|V| = 40K$),
768 hidden size, 12 attention heads

Pre-trained on raw text as a language model

Fine-tuned on labeled data (and language modeling)

Decoder-based Models (GPT)



Decoder-based Models (GPT)

- One job: predict the next word in a sequence
- More formally builds an autoregressive probability model

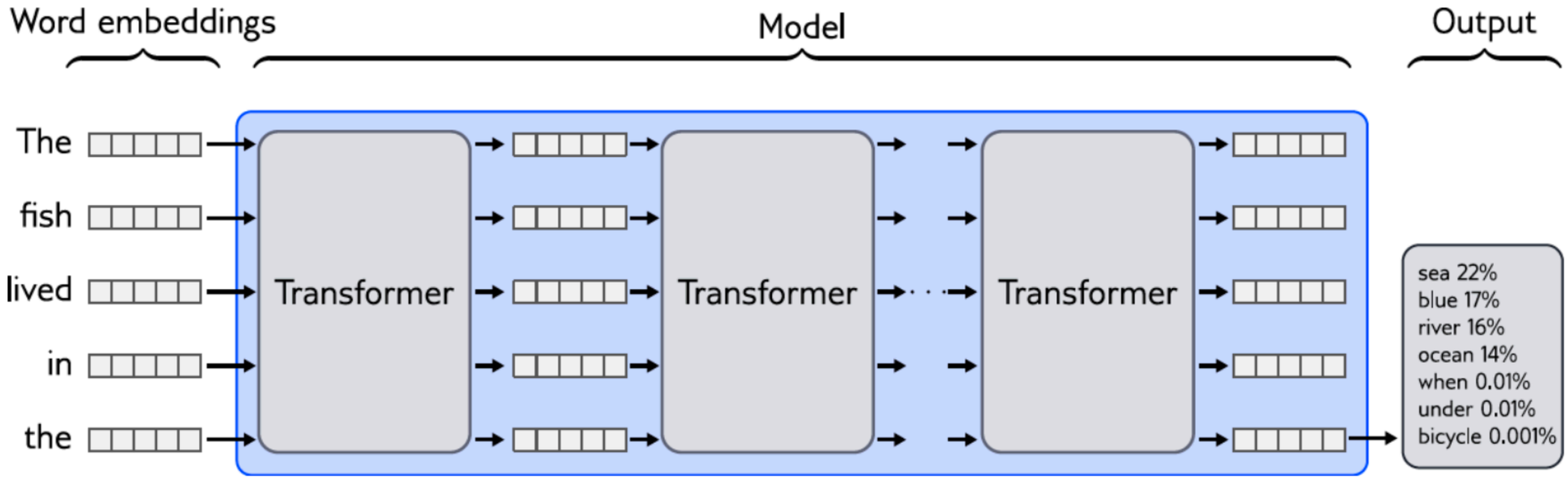
$$Pr(t_1, t_2, \ldots t_N) = Pr(t_1) \prod_{n=2}^{N} Pr(t_n | t_1 \ldots t_{n-1})$$

Decoder-based Models (GPT)

- Builds autoregressive probability model
- E.g. “It takes great courage to let yourself appear weak”

$$
\begin{aligned}
&Pr(\text{It takes great personal courage to let yourself appear weak}) = \\
&Pr(\text{It}) \times Pr(\text{takes}|\text{It}) \times Pr(\text{great}|\text{It takes}) \times Pr(\text{courage}|\text{It takes great}) \times \\
&Pr(\text{to}|\text{It takes great courage}) \times Pr(\text{let}|\text{It takes great courage to}) \times \\
&Pr(\text{yourself}|\text{It takes great courage to let}) \times \\
&Pr(\text{appear}|\text{It takes great courage to let yourself}) \times \\
&Pr(\text{weak}|\text{It takes great courage to let yourself appear}).
\end{aligned}
$$

Decoder-based Models (GPT)



Decoder-based Models (GPT)

GPT 1

Released: 2018

Parameters: 117 Million

Layers: 12

Training Data: Books1 Corpus (7,000 unpublished books)

Focus: Unsupervised pre-training, Transformer architecture, large-scale language modeling

Decoder-based Models (GPT)

GPT 2

Released: 2019

Parameters: ~1.5 Billion

Layers: 48

Training Data: 40GB (English)

Focus: Transformer architecture, self-attention mechanism

Decoder-based Models (GPT)

GPT 3

Released: 2020

Parameters: 175 Billion

Layers: 175

Training Data: 570GB (Multilingual)

Focus: Few-shot learning, prompt engineering, Python support

Decoder-based Models (GPT)

ChatGPT

- GPT3.5 fine-tuned with human annotations
- Trained to predict the next word + be “helpful, honest, harmless”

Prompt:
ELI5: What’s the cause of the "anxiety lump" in our chest during stressful or disheartening experiences?

Labeler demonstration
The änxiety lumpïn your throat is caused by muscular tension keeping your glottis dilated to maximize airflow. The c̈lenched chestör ḧeartachefeeling is caused by the vagus nerve which tells the organs to pump blood faster, stop digesting, and produce adrenaline and cortisol.

Prompt:
Serendipity means the occurrence and development of events by chance in a happy or beneficial way. Use the word in a sentence.

Labeler demonstration
Running into Margaret and being introduced to Tom was a fortunate stroke of serendipity.

Decoder-based Models (GPT)

The Three-Phased Training Approach

1.Supervised Fine-Tuning (SFT):

- Refines a pre-trained GPT-3 model's responses for specific tasks or guidelines, enhancing its understanding and output relevance.

2.Training a Reward Model (RM):

- Develops a system that assesses the quality of text generated by the model, guiding it towards human-preferred responses.

3. Reinforcement Learning from Human Feedback (RLHF)

- Refining AI behavior through direct human feedback.

Decoder-based Models (GPT)

GPT 4

Release: Not Yet

Parameters: ~100 Trillion (speculative)

Layers: Unknown

Training Data: Larger, more diverse (speculative)

Focus: GPT-4 is known to be a multimodal model, capable of processing both text and image inputs to generate text outputs. Advanced few-shot learning, improved NLU and NLG, reasoning and inference

Decoder-based Models (GPT)

Inference

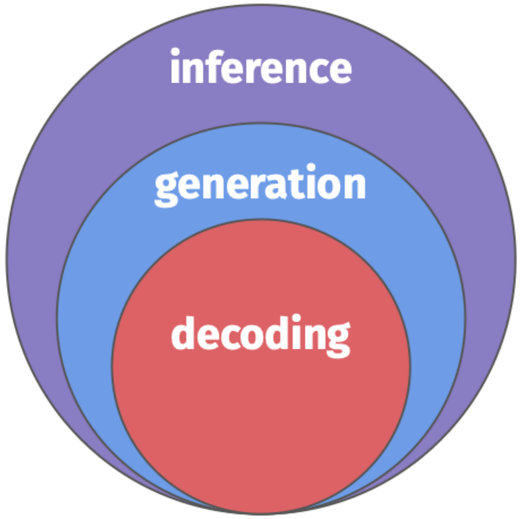
The concept of using a trained model for **making predictions** on new data (for classification, sequence tagging, text generation, ...).

Generation

The process of using a trained model for **producing a sequence of tokens**.

Decoding

The algorithm of **turning the model's internal representation** into a **sequence of tokens**.



Decoder-based Models (GPT)

Task: Generating a **sequence of tokens**.

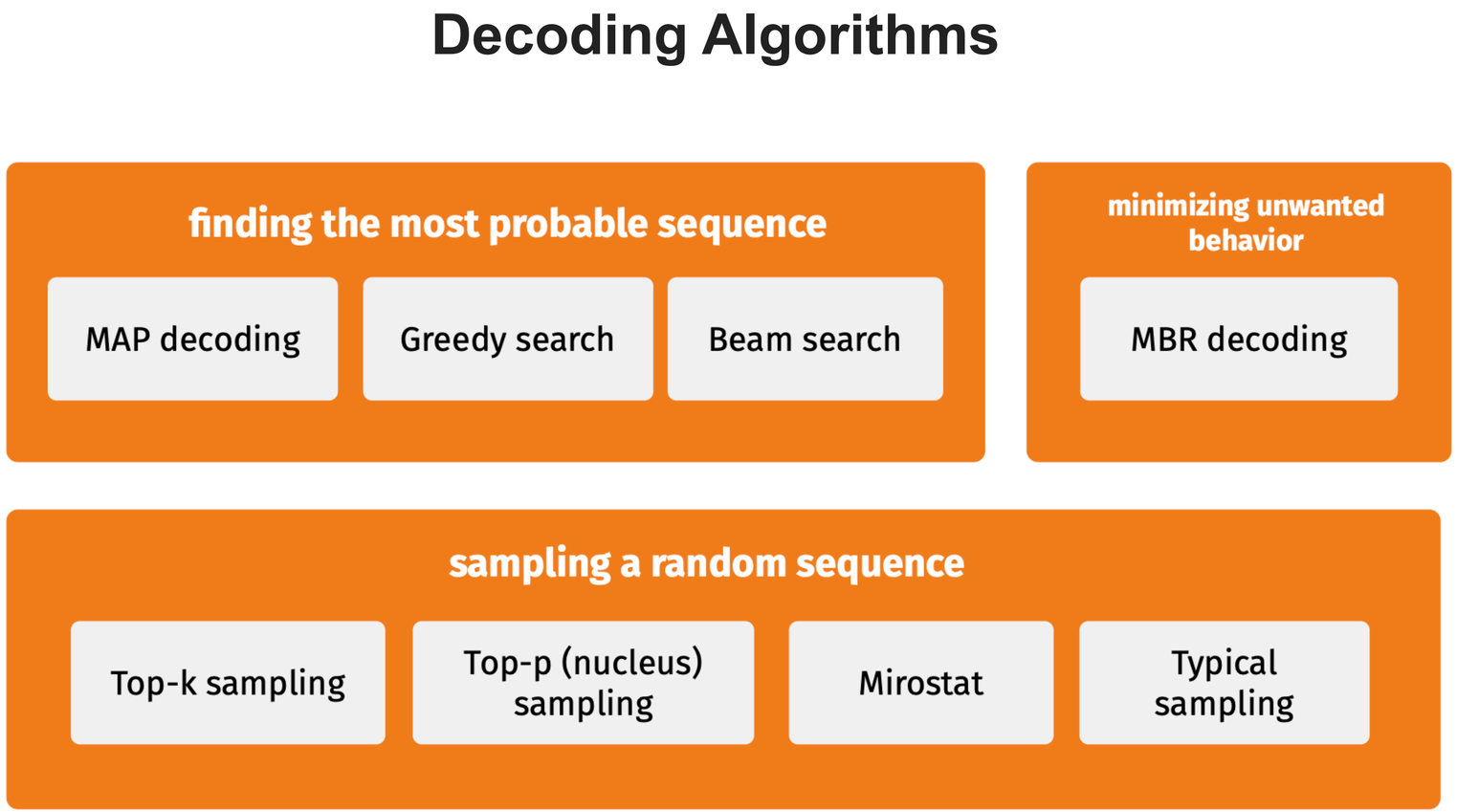
Tool: A language model (LM) giving us a **probability distribution** over the vocabulary for a given prefix.

Method: Feed the sequence prefix in the LM → Select the next token → Append the token to the prefix → Repeat.

(how?)

= Autoregressive decoding

Decoder-based Models (GPT)



Decoder-based Models (GPT)

Exact Inference (Maximum a posteriori decoding)

Finding **the most probable sequence** (=mode of the LM distribution) given the step-wise factorization of sequence probability:

$$y^* = \arg\max_{y \in \mathcal{Y}} P(y) = \arg\max_{y \in \mathcal{Y}} \prod_{i=1}^{t} P(y_i | y_1, \ldots, y_{t-1})$$

Intractable (exponential search space) → approximation algorithms

Decoder-based Models (GPT)

Greedy decoding

Selecting the **most probable token**

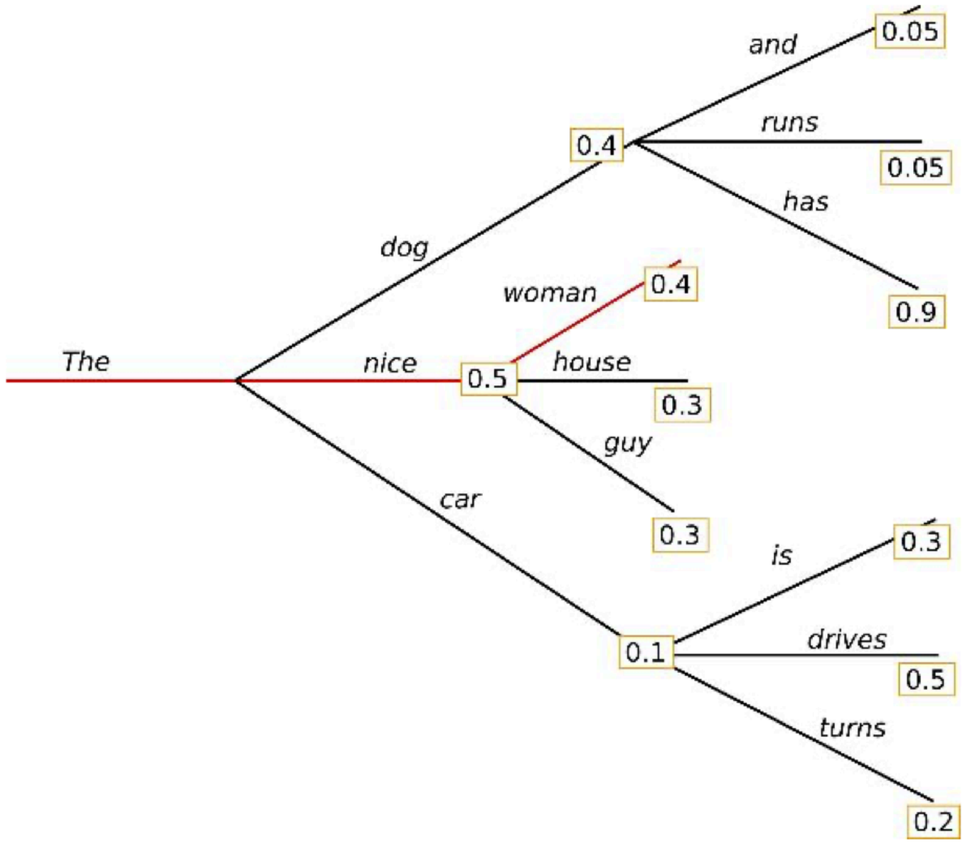
in each step *t:*

$$y_t = \arg\max_{y_t \in \mathcal{V}} P(y_t | y_1, \ldots, y_{t-1})$$

Very fast, often works satisfactorily

(especially with LLMs)

Non-parameteric



Decoder-based Models (GPT)

Beam search

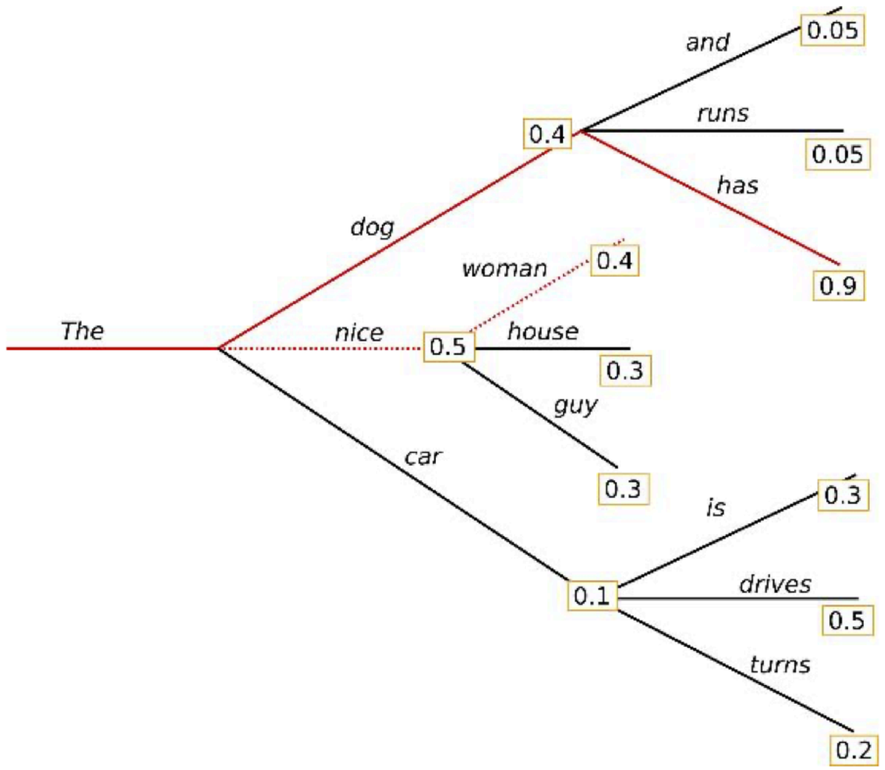
Parameter ***k*: number of sequences**

Each step *t*:

- Extend the sequences from the step *t-1* with all possible tokens.
- Select the *k* most probable sequences for the step *t+1*.

Tuning k:

- *k*=1 == greedy decoding
- larger *k* → slower algorithm
- k>1 allows re-ranking results



Decoder-based Models (GPT)

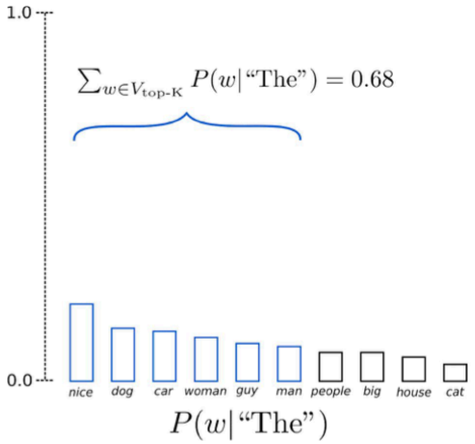
Top-k sampling

Selecting the token in each step **randomly from *k* ∈ {1, ..., |V|} most probable tokens**

The truncated distribution is re-weighed using softmax

The shape of distribution can be adjusted using the **temperature *T***:

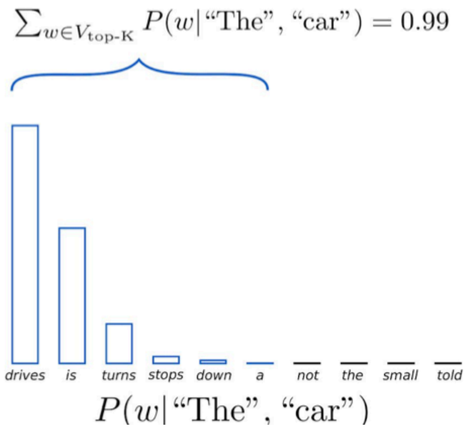
$$\text{softmax}(y_i) = \frac{e^{y_i/T}}{\sum_{y_j \in \mathcal{V}_{\text{top-k}}} e^{y_j/T}}$$



t = 1

prefix = “The” → sampling from {nice, dog, car, woman, guy, man}

cum.prob. = 0.68



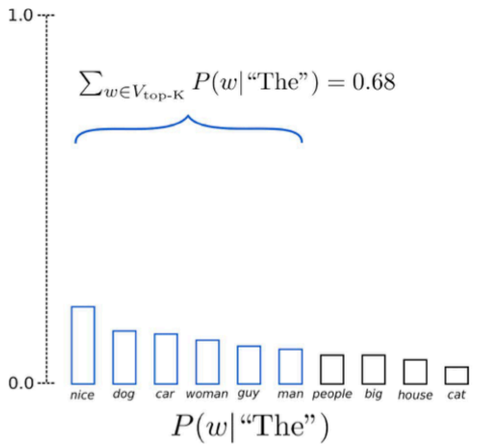
t = 2

prefix = “The car” → sampling from {drives, is, turns, stops, down, a}

cum.prob. = 0.99

Decoder-based Models (GPT)

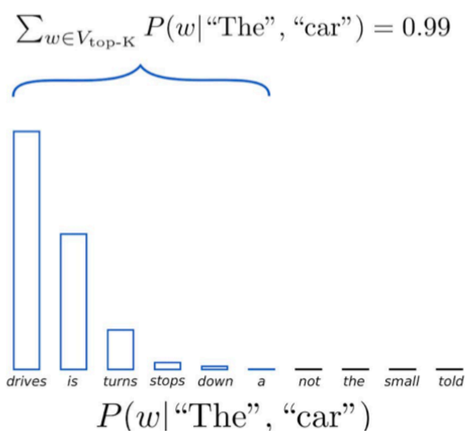
Top-p (nucleus) sampling



t = 1

prefix = “The”
→ sampling from {nice, dog, car, woman, guy, man}

cum.prob. = 0.68



t = 2

prefix = “The car”
→ sampling from {drives, is, turns, stops, down, a}

cum.prob. = 0.99

Specifies the cumulative probability score threshold that the tokens must reach. If p=0.69, the result is the same as top-k

Encoder-Decoder Models (T5)

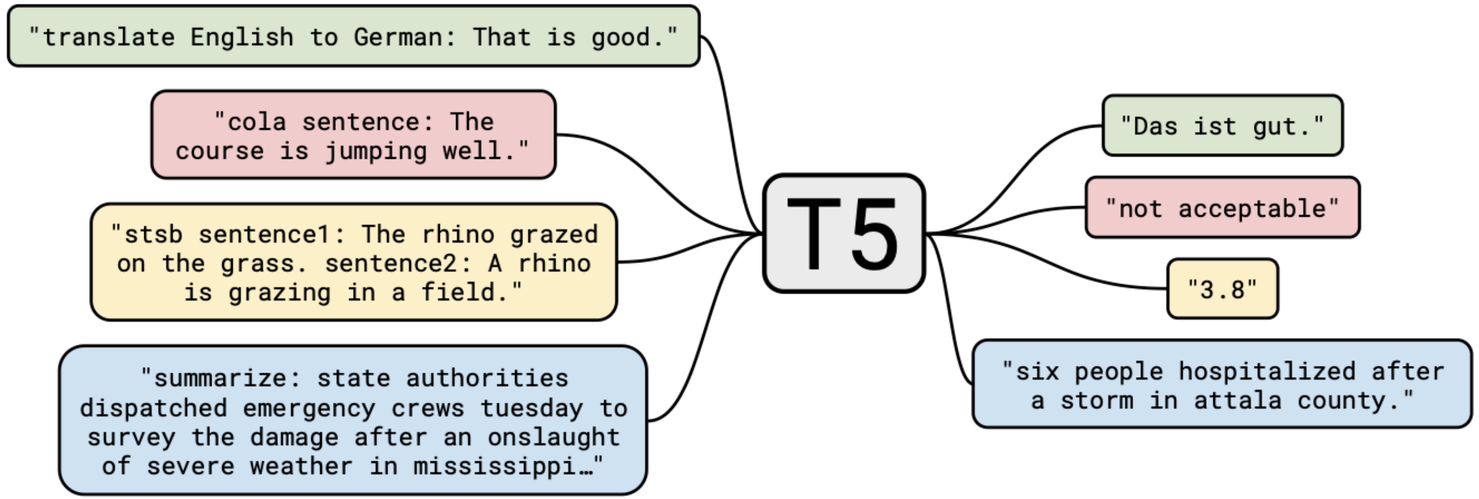


Figure 1: A diagram of our text-to-text framework. Every task we consider—including translation, question answering, and classification—is cast as feeding our model text as input and training it to generate some target text. This allows us to use the same model, loss function, hyperparameters, etc. across our diverse set of tasks. It also provides a standard testbed for the methods included in our empirical survey. "T5" refers to our model, which we dub the "**T**ext-**t**o-**T**ext **T**ransfer **T**ransformer".

Raffel, C., Shazeer, N., Roberts, A., Lee, K., Narang, S., Matena, M., ... & Liu, P. J. (2020). Exploring the limits of transfer learning with a unified text-to-text transformer. Journal of machine learning research, 21(140), 1-67.

Encoder-Decoder Models (T5)

Operates on an encoder-decoder model.

Pretrained on a multi-task mixture of both unsupervised and supervised tasks, each converted into a text-to-text format.

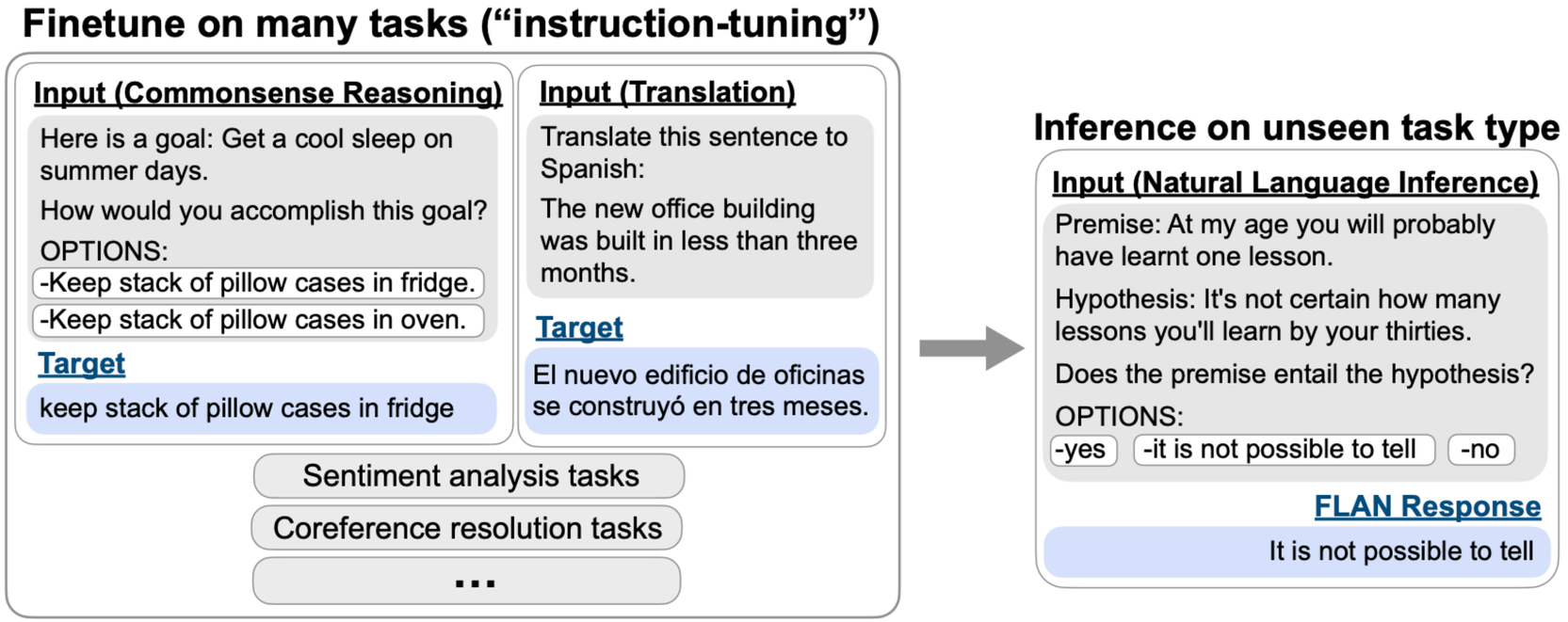
Key Focus: Transfer learning - Pre-training on data-rich tasks followed by fine-tuning on downstream tasks.

Adapts to various tasks by prepending task-specific prefixes to the input, e.g., for translation: "translate English to German: ...", for summarization: "summarize: ...".

Instruction Finetuning (FLAN)

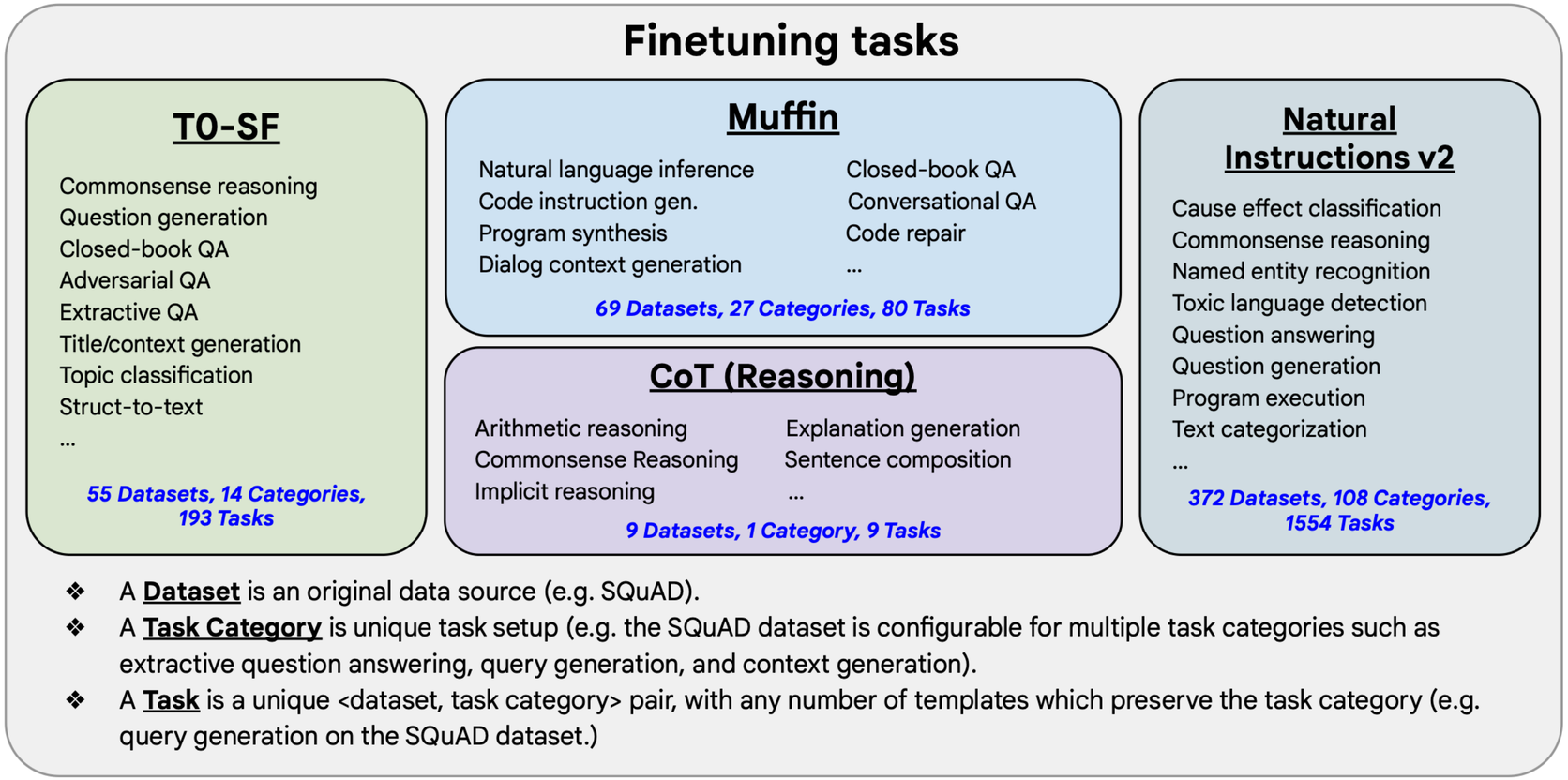
Father of Instruction-Finetuning

FLAN (Finetuning language models)



Wei, J., Bosma, M., Zhao, V. Y., Guu, K., Yu, A. W., Lester, B., ... & Le, Q. V. (2021). Finetuned language models are zero-shot learners. arXiv preprint arXiv:2109.01652.

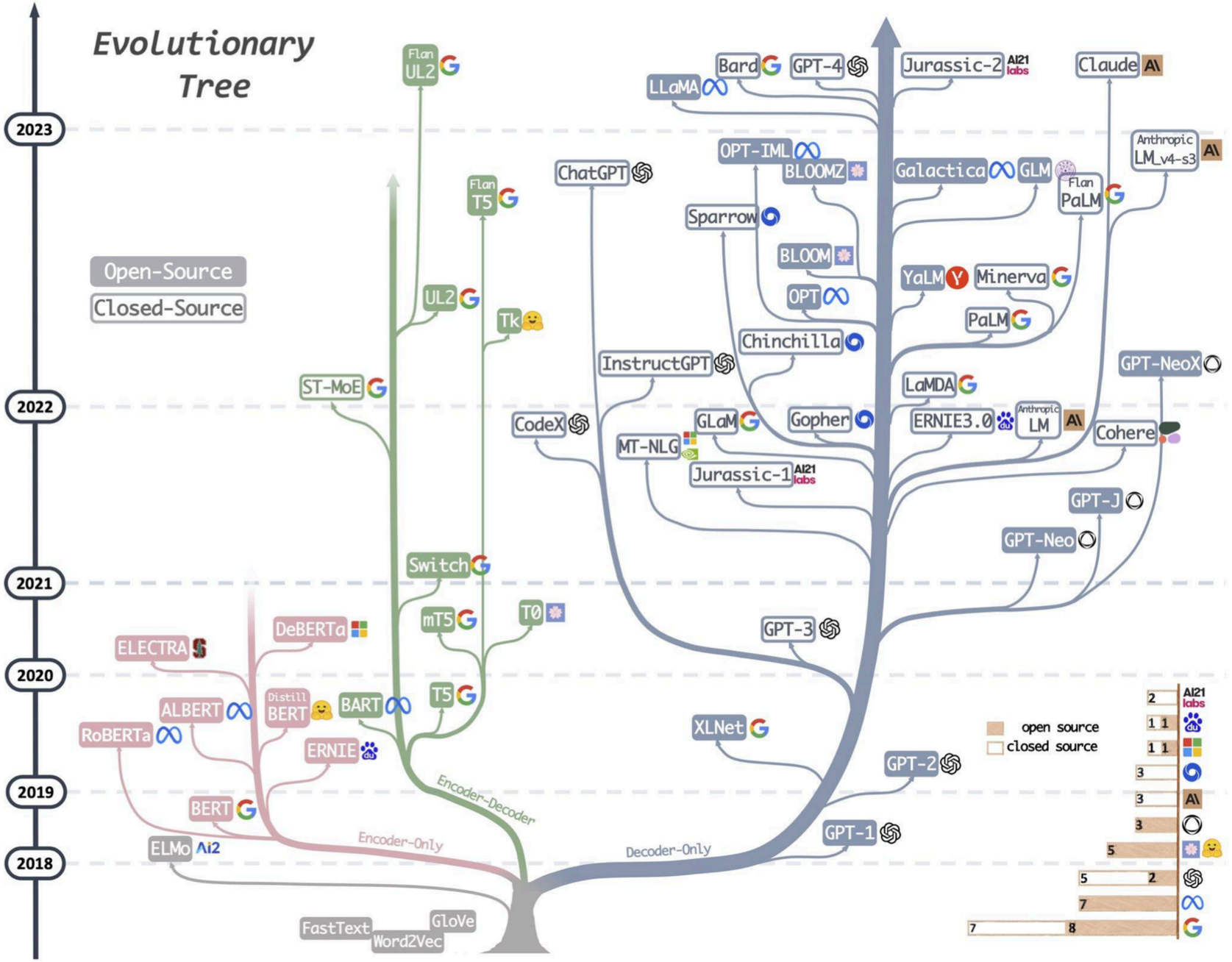
Encoder-Decoder Models (Flan-T5)



Raffel, C., Shazeer, N., Roberts, A., Lee, K., Narang, S., Matena, M., ... & Liu, P. J. (2020). Exploring the limits of transfer learning with a unified text-to-text transformer. Journal of machine learning research, 21(140), 1-67.

Few-shot Learning and Prompts

	No Prompt	Prompt
Zero-shot (0s)	skicts = sticks	Please unscramble the letters into a word, and write that word: skicts = sticks
1-shot (1s)	chiar = chair skicts = sticks	Please unscramble the letters into a word, and write that word: chiar = chair skicts = sticks
Few-shot (FS)	chiar = chair [...] pciinc = picnic skicts = sticks	Please unscramble the letters into a word, and write that word: chiar = chair [...] pciinc = picnic skicts = sticks


Evolutionary Tree
2023
2022
2021
2020
2019
2018
Open-Source
Closed-Source
Flan UL2
Flan T5
UL2
Tk
ST-MoE
Switch
mT5
T0
DeBERTa
ELECTRA
ALBERT
Distill BERT
BART
T5
RoBERTa
ERNIE
BERT
ELMo
Ai2
FastText
GloVe
Word2Vec
Encoder-Only
Encoder-Decoder
Decoder-Only
ChatGPT
Bard
GPT-4
LLaMA
OPT-IML
BLOOMZ
Sparrow
BLOOM
OPT
Chinchilla
InstructGPT
CodeX
GLaM
Gopher
MT-NLG
Jurassic-1
GPT-3
XLNet
GPT-2
GPT-1
Jurassic-2
Claude
Anthropic LM_v4-s3
Galactica
GLM
Flan PaLM
YaLM
Minerva
PaLM
LaMDA
ERNIE3.0
Anthropic LM
GPT-NeoX
Cohere
GPT-J
GPT-Neo
open source
closed source
2
1 1
1 1
3
3
3
5
5 2
7
7 8

ORTA DOĞU TEKNİK ÜNİVERSİTESİ
MIDDLE EAST TECHNICAL UNIVERSITY

Thanks for your participation!

Çağrı Toraman
16.12.2025