Приложения моделей Transformer

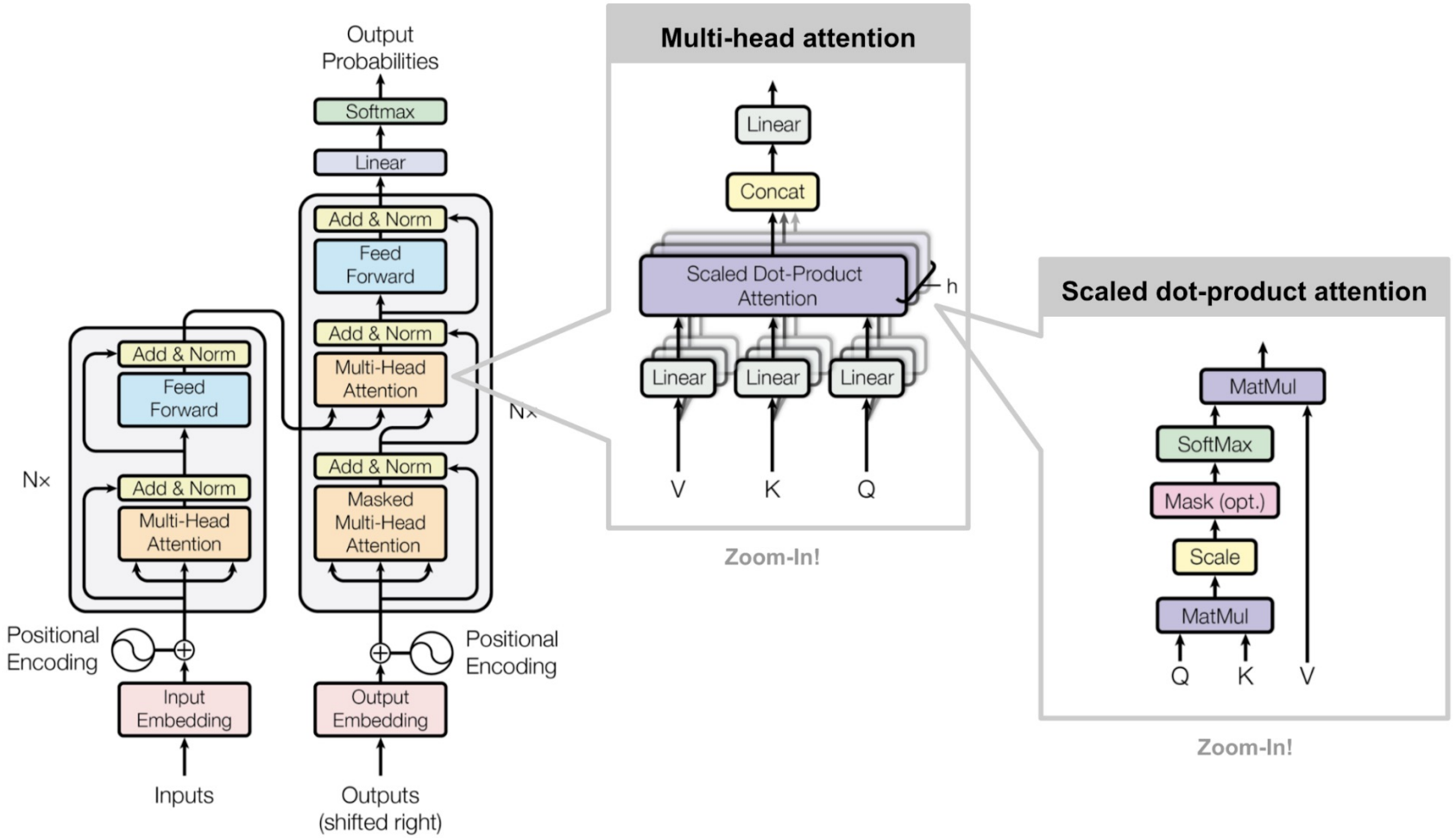
Кафедра технологий проектирования сложных технических систем

2023

План лекции

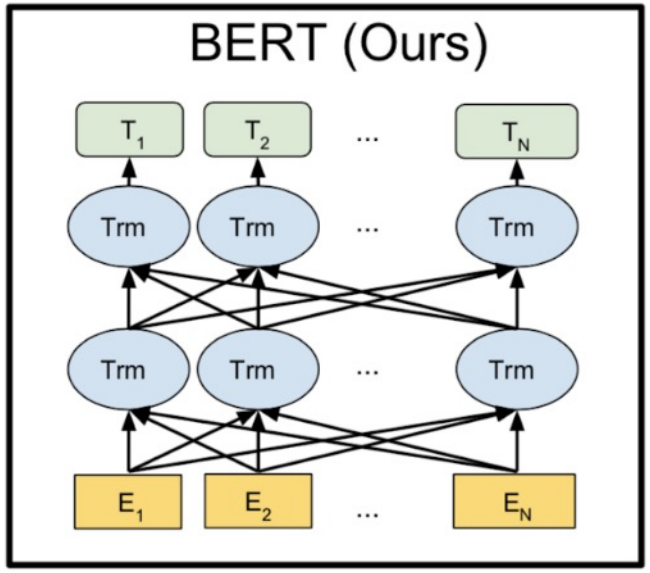
- Attention (повторение)
- CLIP framework
  - Transformer for images
- Question answering (QA)
  - Text generation
  - Close & Open domain

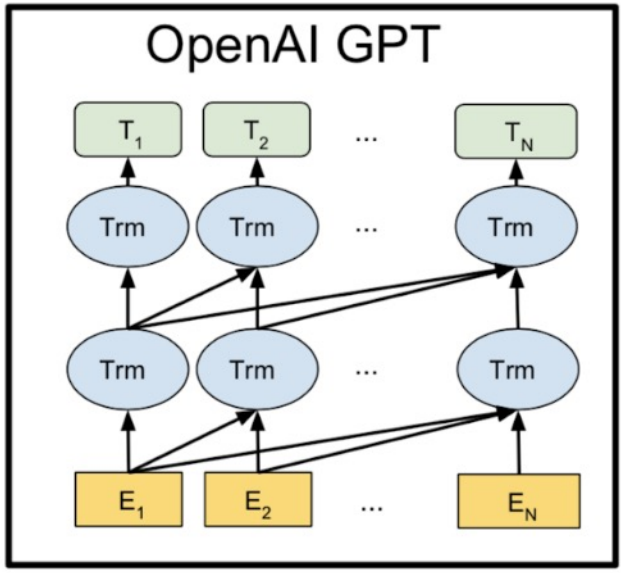
Transformer

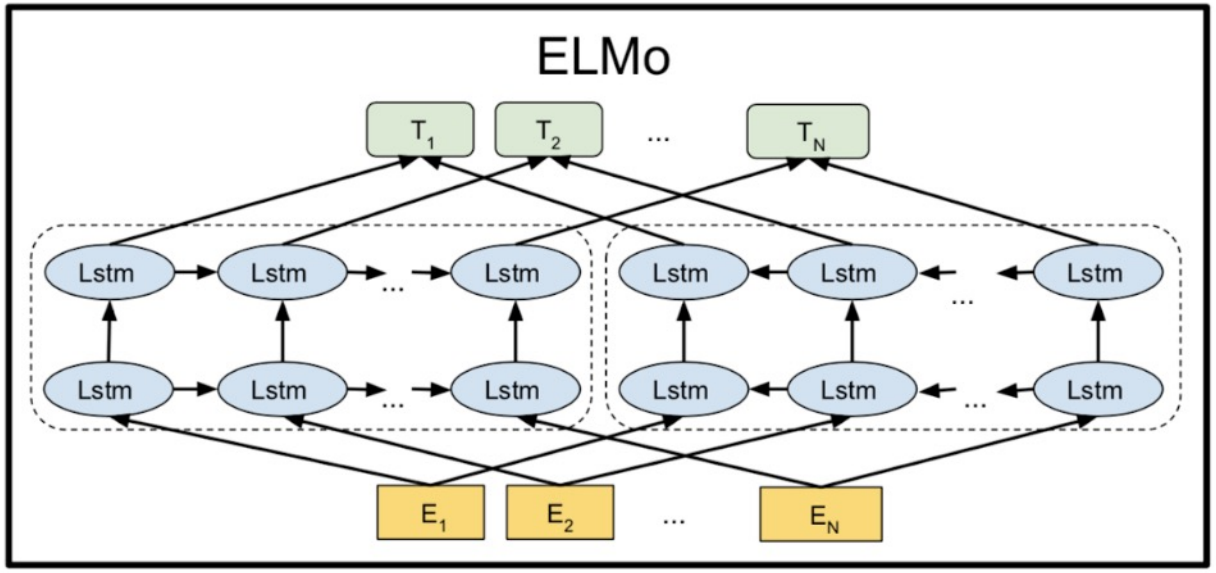


Attention Is All You Need Vaswani, et al., 2017

Transformer basis







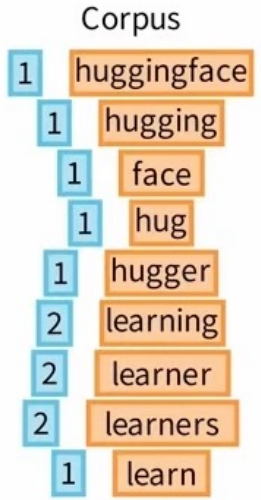
Bidirectional Encoder Representations from Transformers (Devlin, et al., 2019)

Byte Pair Encoding

this is the hugging face course. this chapter is about tokenization. this section shows several tokenizer algorithms.

this | is | the | hugging | face | course | . | this | chapter | is | about | tokenization | . | this | section | shows | several | tokenizer | algorithms | .

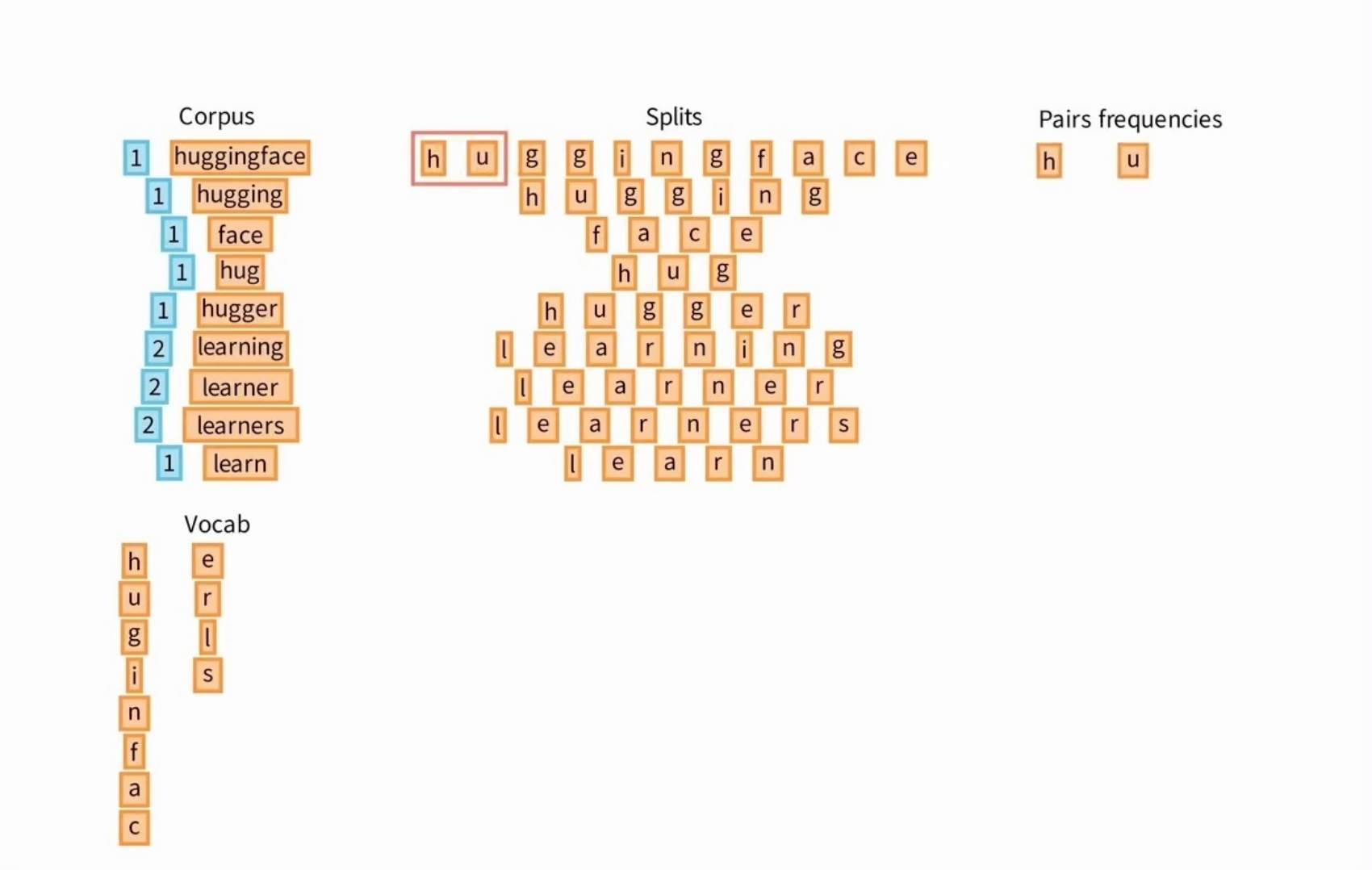
3x this | 2x is | 1x the | 1x hugging | 1x face | 1x course | 3x . | 1x chapter | 1x about | 1x tokenization | 1x section | 1x shows | 1x several | 1x tokenizer | 1x algorithms



Byte Pair Encoding (BPE)

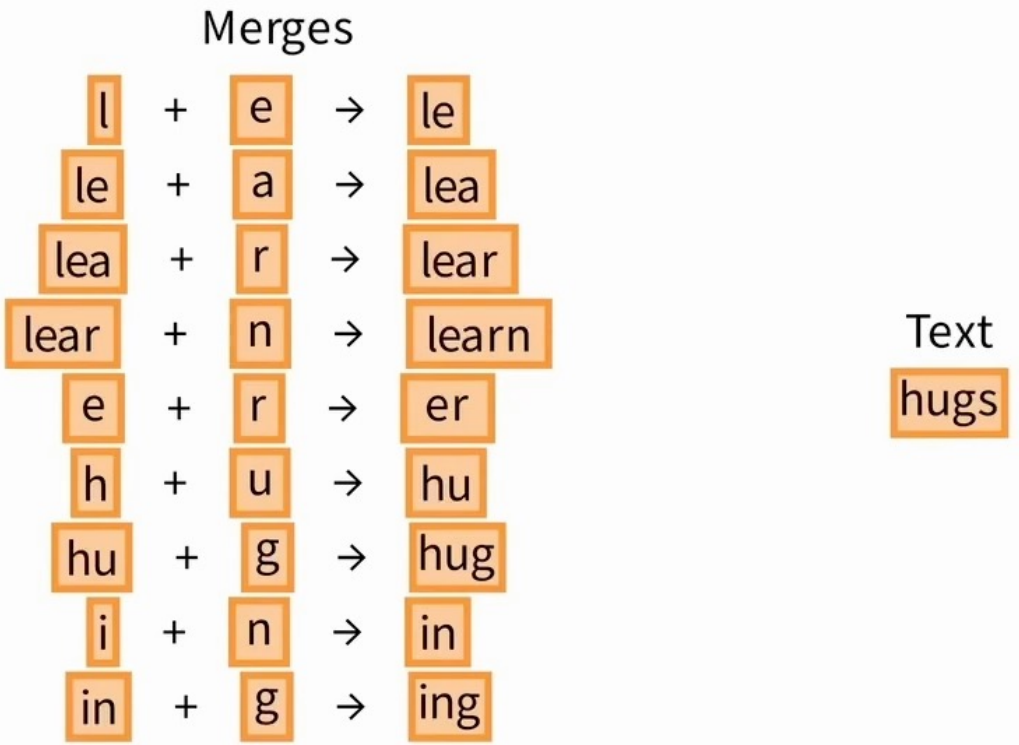
Byte-Pair Encoding tokenization (HuggingFace)

Byte Pair Encoding



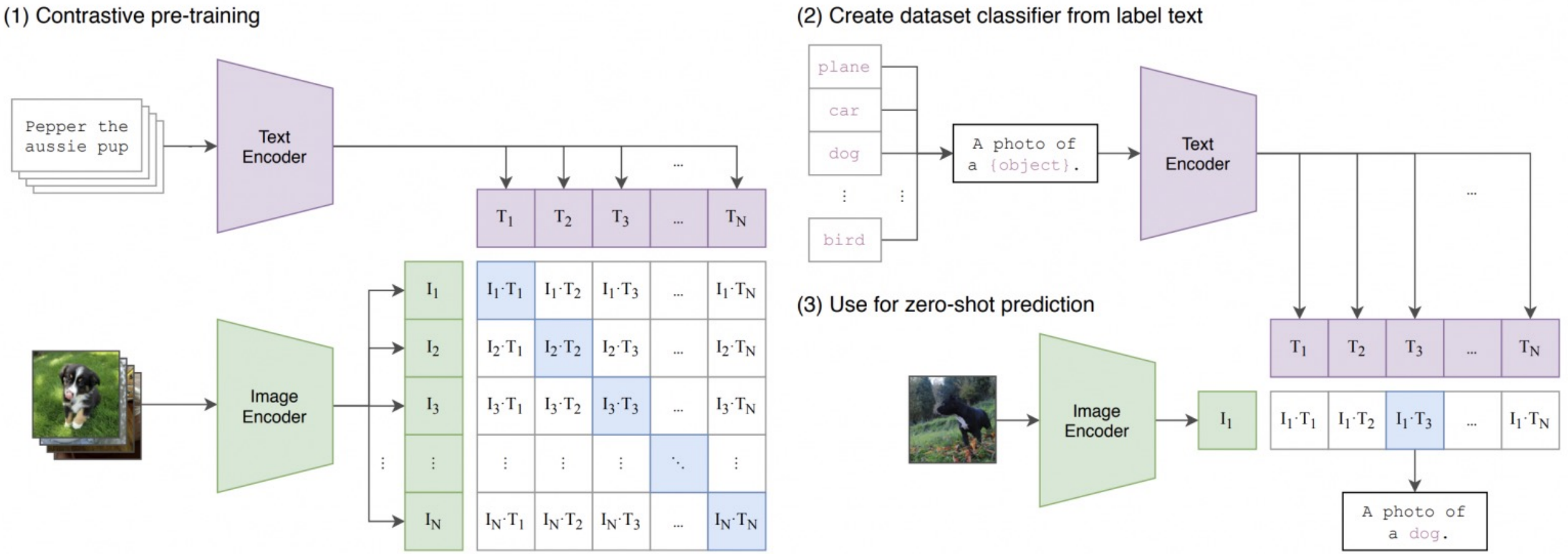
Byte-Pair Encoding tokenization (HuggingFace)

Byte Pair Encoding

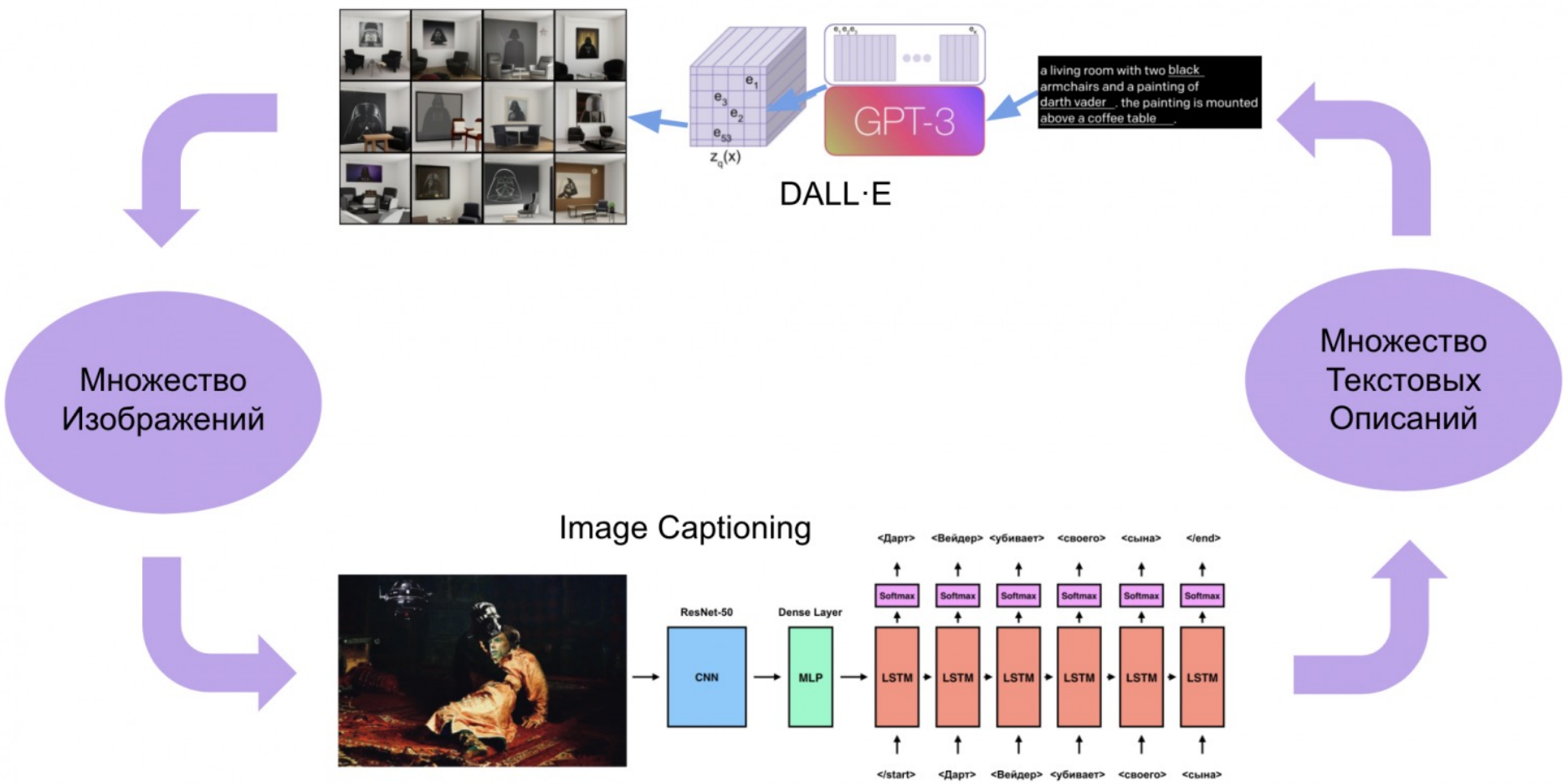


Byte-Pair Encoding tokenization (HuggingFace)

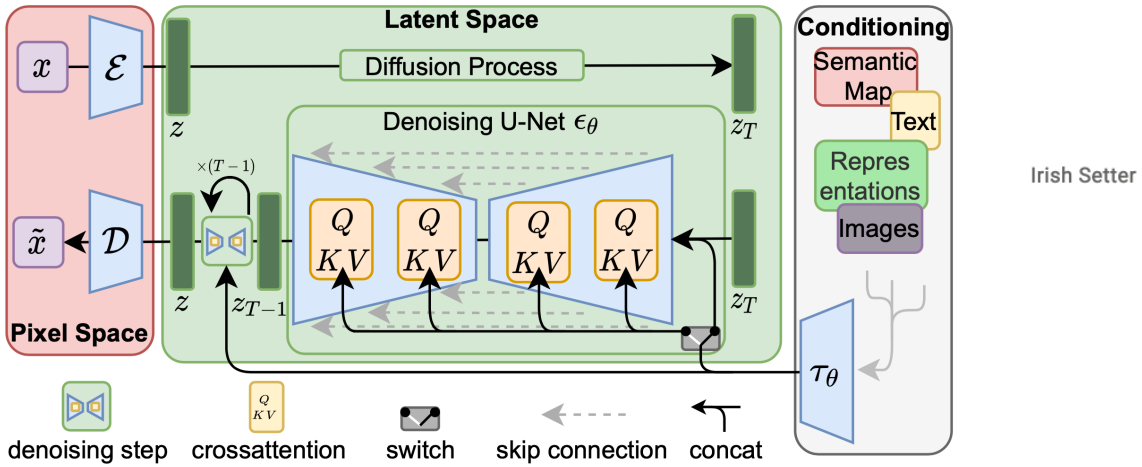
CLIP


(1) Contrastive pre-training
Pepper the aussie pup
Text Encoder
Image Encoder
T1 T2 T3 ... TN
I1 I2 I3 IN
I1·T1 I1·T2 I1·T3 ... I1·TN
I2·T1 I2·T2 I2·T3 ... I2·TN
I3·T1 I3·T2 I3·T3 ... I3·TN
IN·T1 IN·T2 IN·T3 ... IN·TN
(2) Create dataset classifier from label text
plane
car
dog
bird
A photo of a {object}.
Text Encoder
T1 T2 T3 ... TN
(3) Use for zero-shot prediction
Image Encoder
I1
I1·T1 I1·T2 I1·T3 ... I1·TN
A photo of a dog.

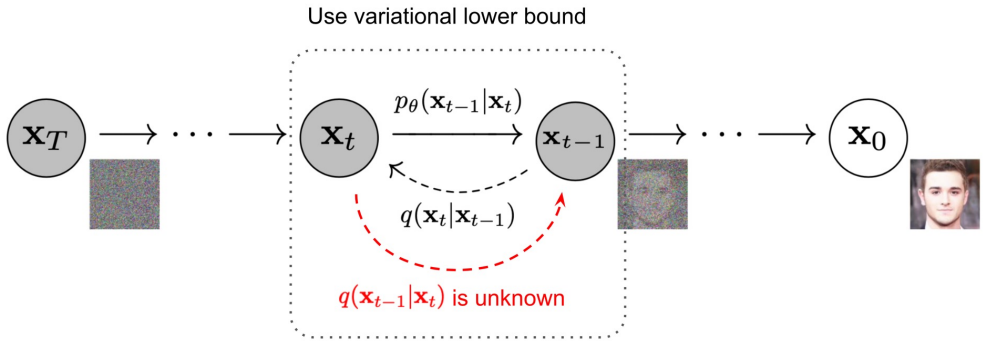
CLIP + Image generation



Diffusion models (recap)



Irish Setter



[Больше математики про diffusion]

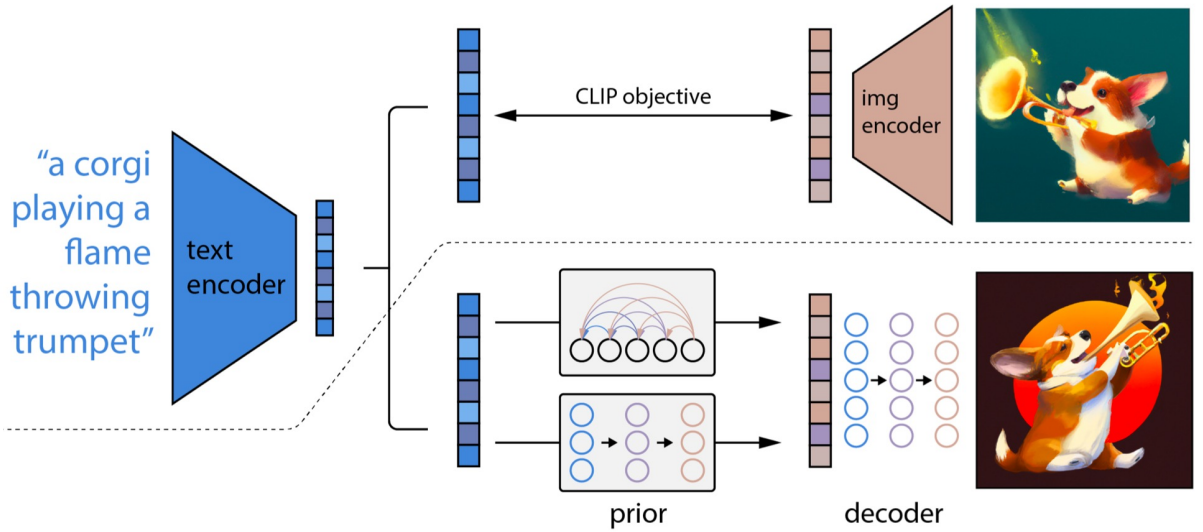
Promt Image Generation

The two-stage diffusion model **unCLIP** (Ramesh et al. 2022) heavily utilizes the CLIP text encoder to produce text-guided images at high quality. Given a pretrained CLIP model $\mathbf{c}$ and paired training data for the diffusion model, $(\mathbf{x}, y)$, where x is an image and y is the corresponding caption, we can compute the CLIP text and image embedding, $\mathbf{c}^t(y)$ and $\mathbf{c}^i(\mathbf{x})$, respectively. The unCLIP learns two models in parallel:

- A prior model $P(\mathbf{c}^i|y)$: outputs CLIP image embedding $\mathbf{c}^i$ given the text y.
- A decoder $P(\mathbf{x}|\mathbf{c}^i, [y])$: generates the image $\mathbf{x}$ given CLIP image embedding $\mathbf{c}^i$ and optionally the original text y.

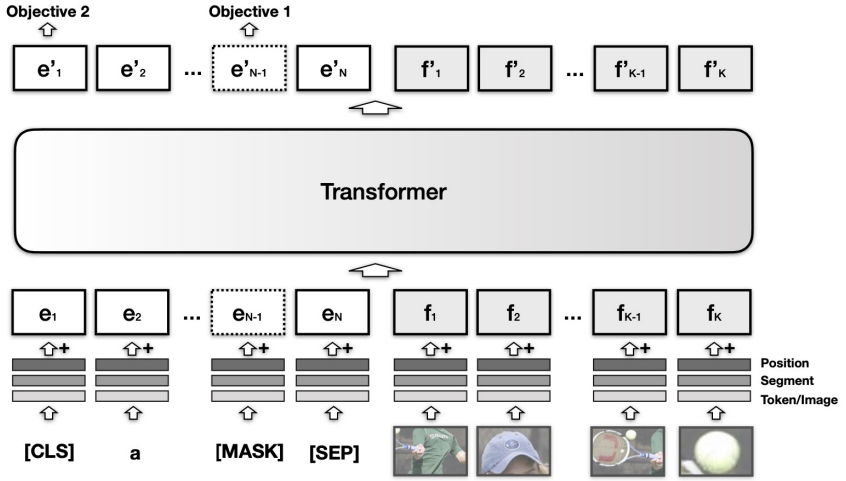
unCLIP follows a two-stage image generation process:

1. Given a text y, a CLIP model is first used to generate a text embedding $\mathbf{c}^t(y)$. Using CLIP latent space enables zero-shot image manipulation via text.
2. A diffusion or autoregressive prior $P(\mathbf{c}^i|y)$ processes this CLIP text embedding to construct an image prior and then a diffusion decoder $P(\mathbf{x}|\mathbf{c}^i, [y])$ generates an image, conditioned on the prior. This decoder can also generate image variations conditioned on an image input, preserving its style and semantics.

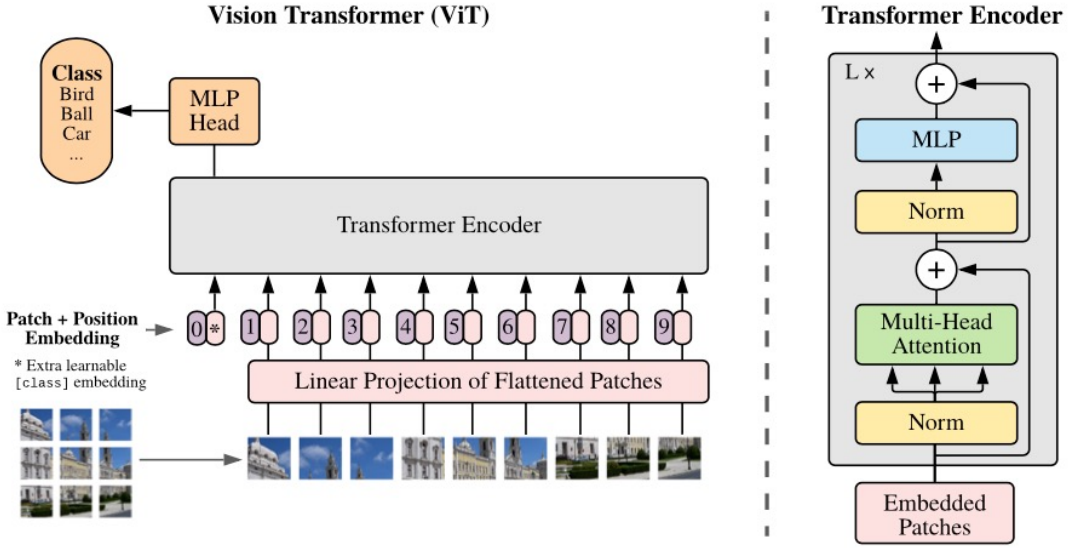


unCLIP (Ramesh et al. 2022)
Lilian Weng What are Diffusion Models?

Visual Transformers



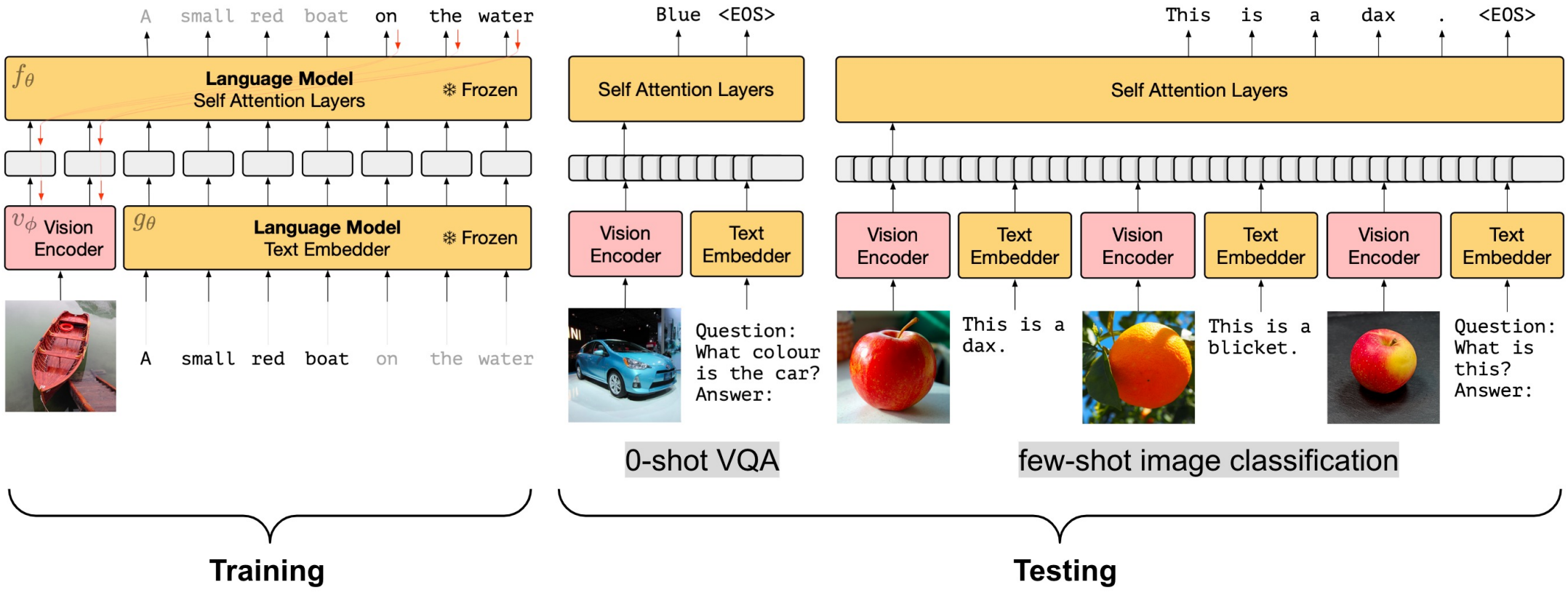
A person hits a ball with a tennis racket



VisualBERT (Li et al. 2019)

An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale (Alexey et al. 2020)

Learned Image Embedding

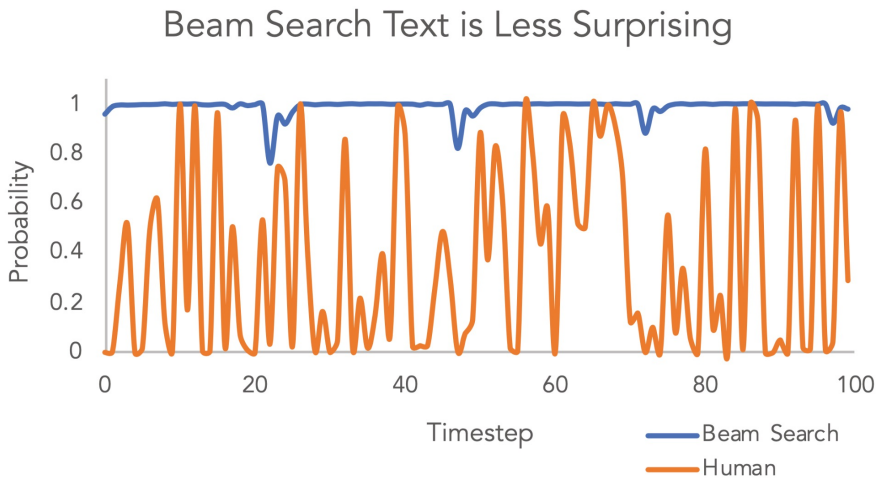


Multimodal Few-Shot Learning with Frozen Language Models (Tsimpoukelli et al. 2021)

Question & Answering

Text generation

$$p_i \propto \frac{\exp(o_i/T)}{\sum_j \exp(o_j/T)}$$



Guided Decoding

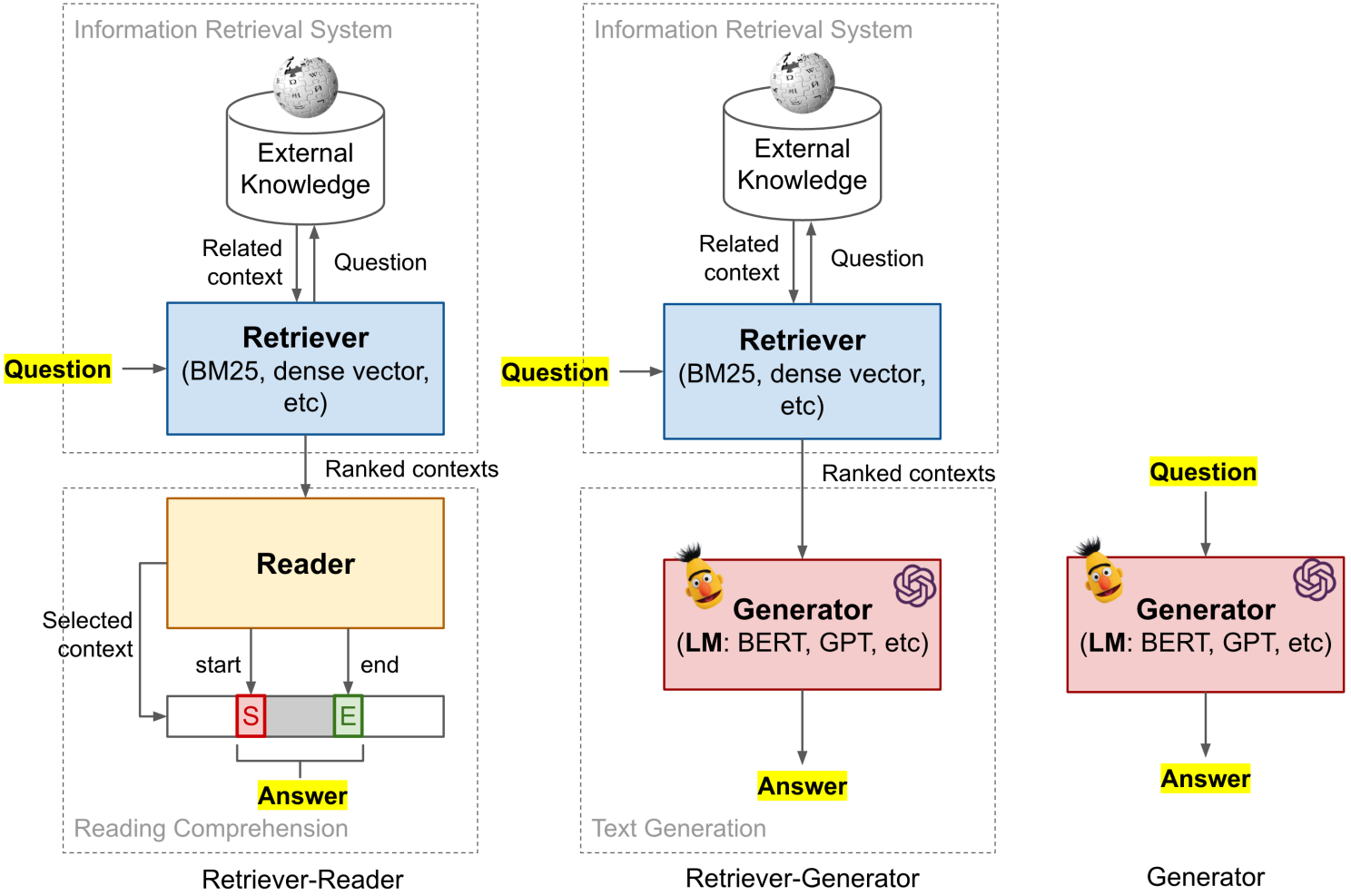
$$\mathbf{y}^* = \arg\max_{\mathbf{y} \in \mathcal{Y}} \big(\underbrace{\log p_\theta(\mathbf{y}|\mathbf{x})}_{\text{MAP}} - \underbrace{\lambda \mathcal{R}(\mathbf{y})}_{\text{regularizer}} \big)$$

1. *Greedy*: $\mathcal{R}_{\text{greedy}}(\mathbf{y}) = \sum_{t=1}^{|\mathbf{y}|} \big(u_t(y_t) - \min_{y' \in \mathcal{V}} u_t(y')\big)^2$; if set $\lambda \to \infty$, we have greedy search. Note that being greedy at each individual step does not guarantee global optimality.
2. *Variance regularizer*: $\mathcal{R}_{\text{var}}(\mathbf{y}) = \frac{1}{|\mathbf{y}|} \sum_{t=1}^{|\mathbf{y}|} \big(u_t(y_t) - \bar{u}\big)^2$, where $\bar{u}$ is the average surprisal over all timesteps. It directly encodes the UID hypothesis.
3. *Local consistency*: $\mathcal{R}_{\text{local}}(\mathbf{y}) = \frac{1}{|\mathbf{y}|} \sum_{t=1}^{|\mathbf{y}|} \big(u_t(y_t) - u_{t-1}(y_{t-1})\big)^2$; this decoding regularizer encourages adjacent tokens to have similar surprisal.
4. *Max regularizer*: $\mathcal{R}_{\text{max}}(\mathbf{y}) = \max_t u_t(y_t)$ penalizes the maximum compensation of surprisal.
5. *Squared regularizer*: $\mathcal{R}_{\text{square}}(\mathbf{y}) = \sum_{t=1}^{|\mathbf{y}|} u_t(y_t)^2$ encourages all the tokens to have surprisal close to 0.

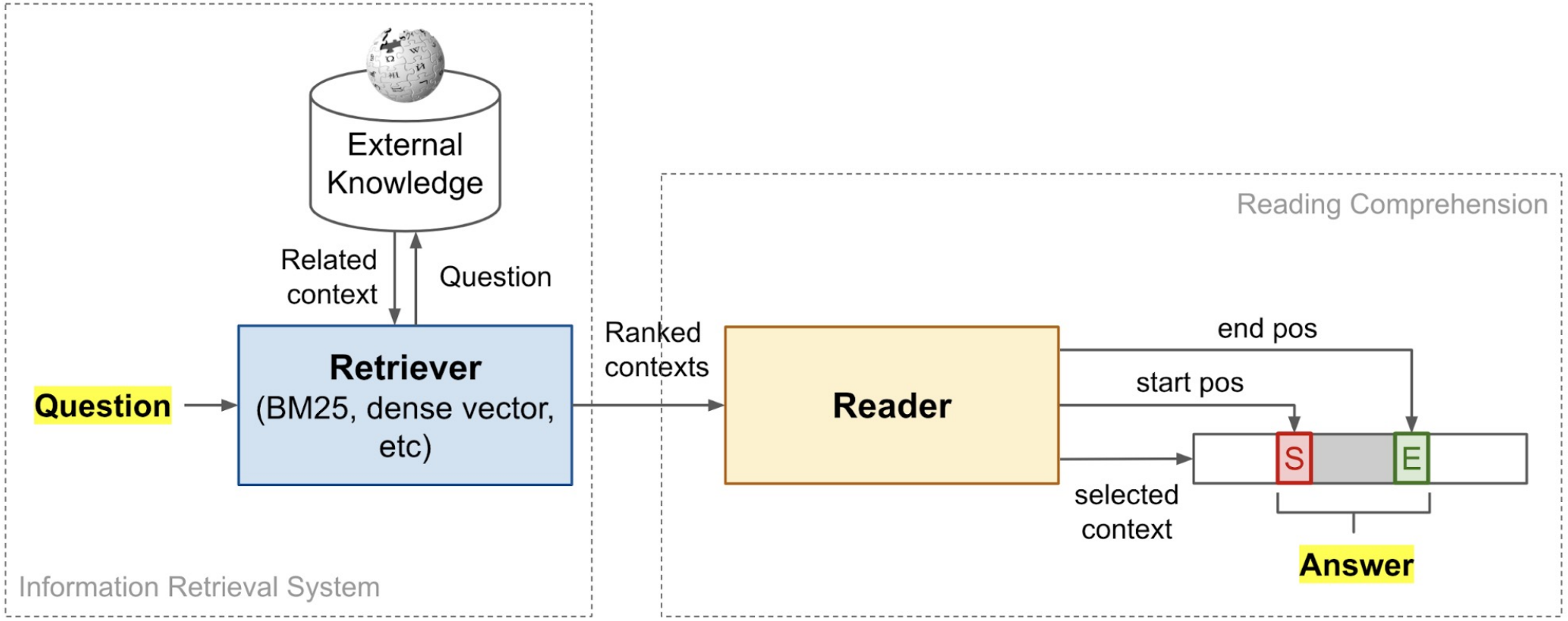
Greedy search
Beam search
Top-k sampling

> "The uniform information density hypothesis (UID; Levy and Jaeger, 2007) states that—subject to the constraints of the grammar—humans prefer sentences that distribute information (in the sense of information theory) equally across the linguistic signal, e.g., a sentence."

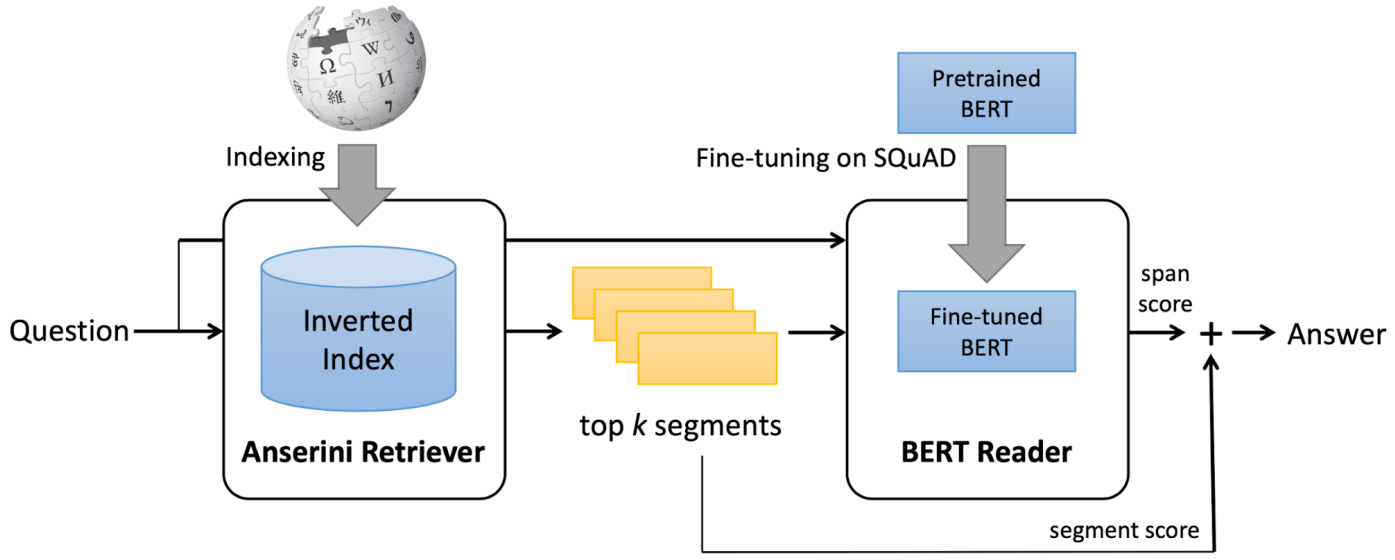
Question & Answering



Open-book QA: Retriever-Reader



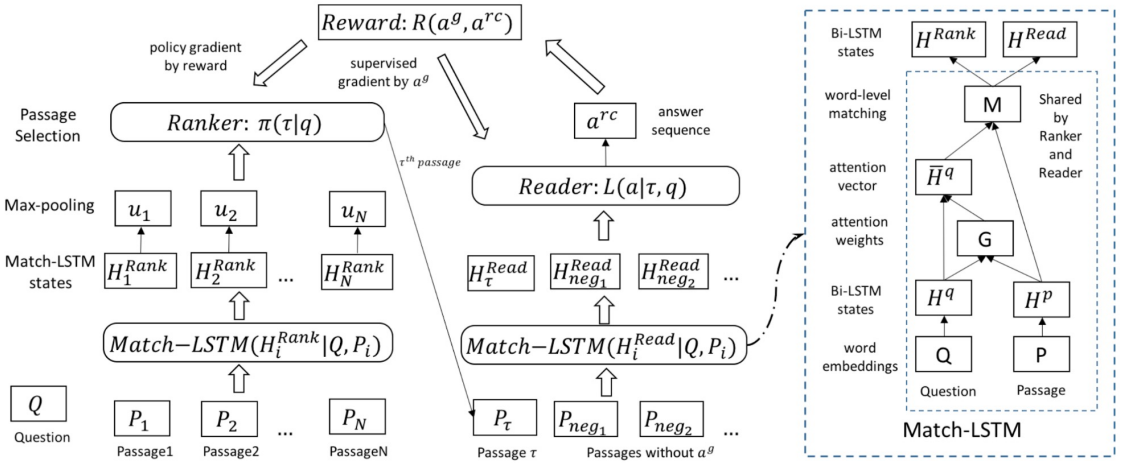
Open-book QA: Retriever-Reader

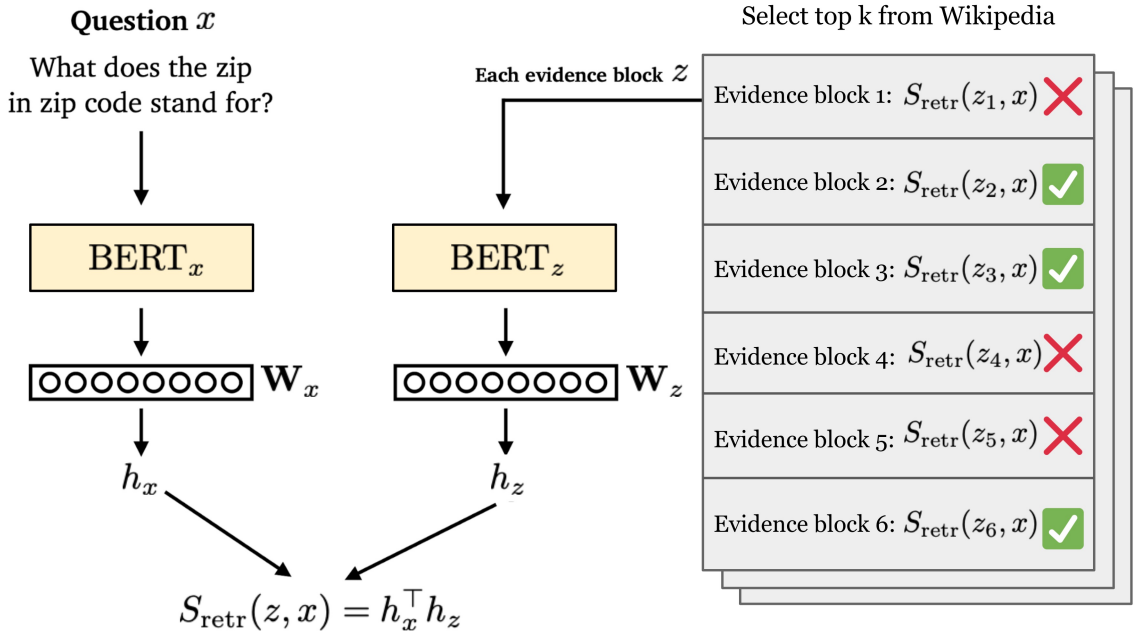


$$\text{tf-idf}(t, d, \mathcal{D}) = \text{tf}(t, d) \times \text{idf}(t, \mathcal{D})$$
$$\text{tf}(t, d) = \log(1 + \text{freq}(t, d))$$
$$\text{idf}(t, \mathcal{D}) = \log\left(\frac{|\mathcal{D}|}{|d \in \mathcal{D} : t \in d|}\right)$$

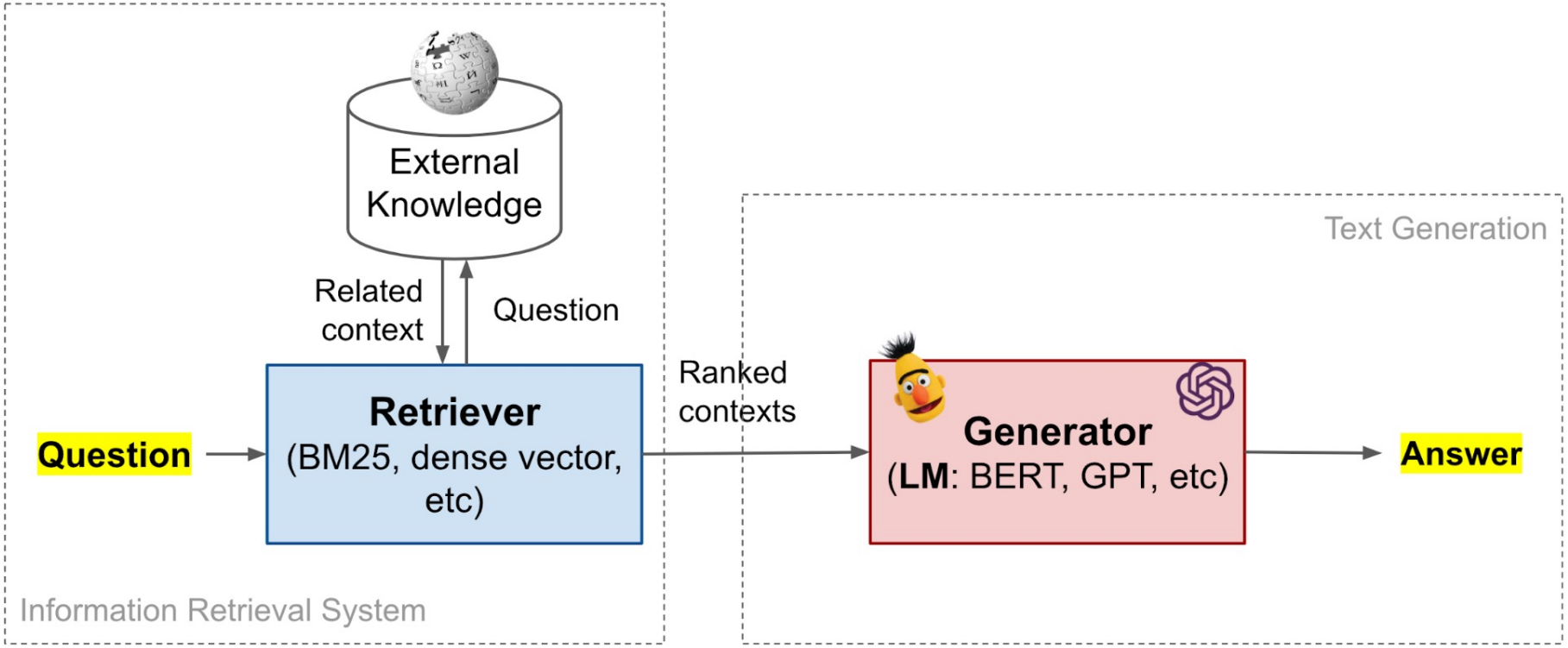
Open-book QA: Retriever-Reader

End to end training



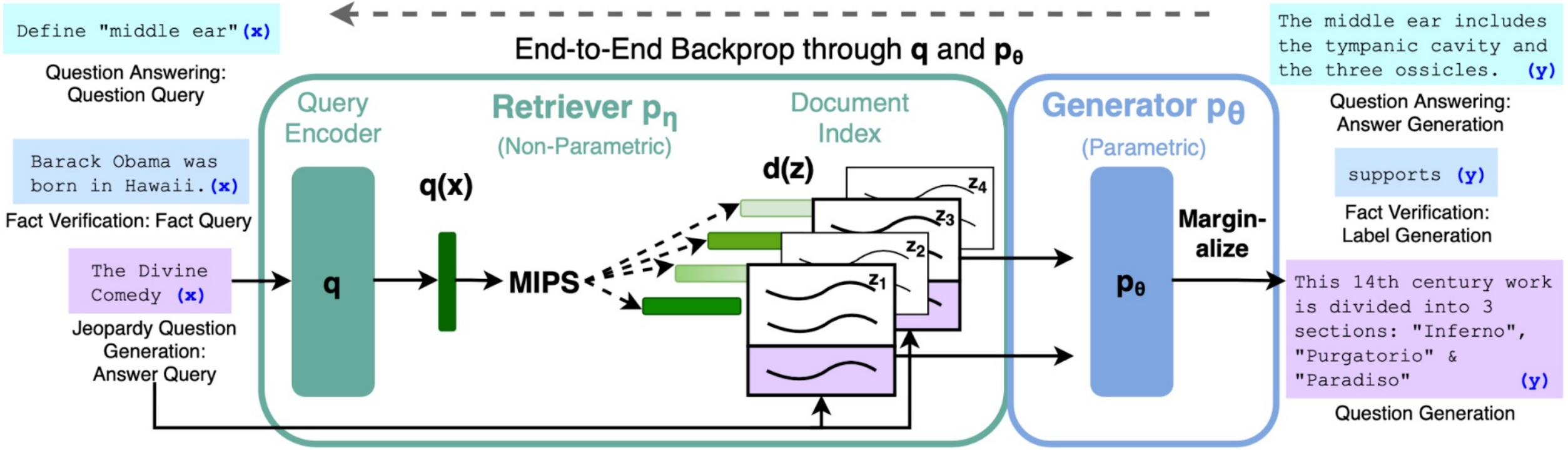


Open-book QA: Retriever-Generator



Open-book QA: Retriever-Generator

End to end training



Заключение

- Attention (повторение)
- CLIP framework
  - Transformer for images
- Question answering (QA)
  - Text generation
  - Close & Open domain