Technische Universität Braunschweig

Institut für Nachrichtentechnik

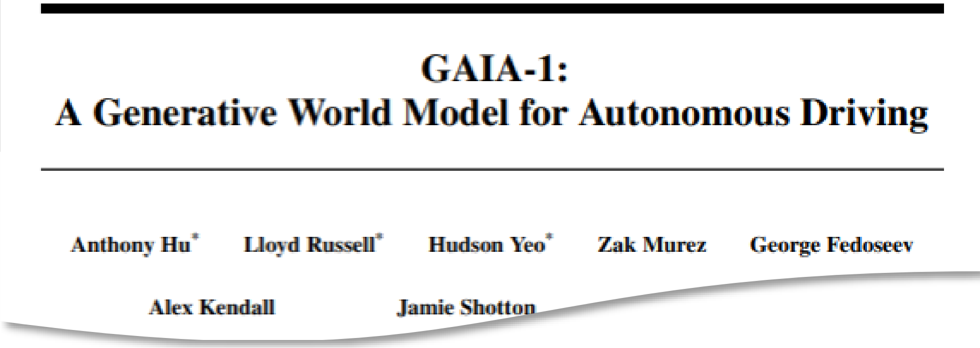
From ChatGPT to GAIA-1:
On Generative Sequence Models in Speech, Language, and Vision

Tim Fingscheidt, Timo Lohrenz, Zhengyang Li, Björn Möller

GAIA-1
New possibilities in autonomous driving R&D

What can GAIA-1 do?

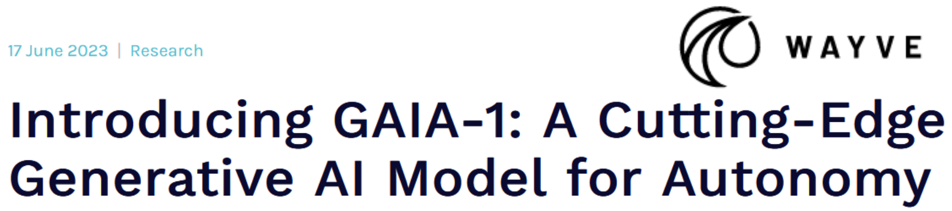
- It allows multimodal prompting with a video, text, and action, and …
- … hallucinates a realistic continuation of the video, under text and action constraints



GAIA-1: A Generative World Model for Autonomous Driving

Anthony Hu* Lloyd Russell* Hudson Yeo* Zak Murez George Fedoseev
Alex Kendall Jamie Shotton

[1] Hu, Anthony, et al. "GAIA-1: A Generative World Model for Autonomous Driving." *arXiv preprint arXiv:2309.17080* (2023).



17 June 2023 | Research

WAYVE

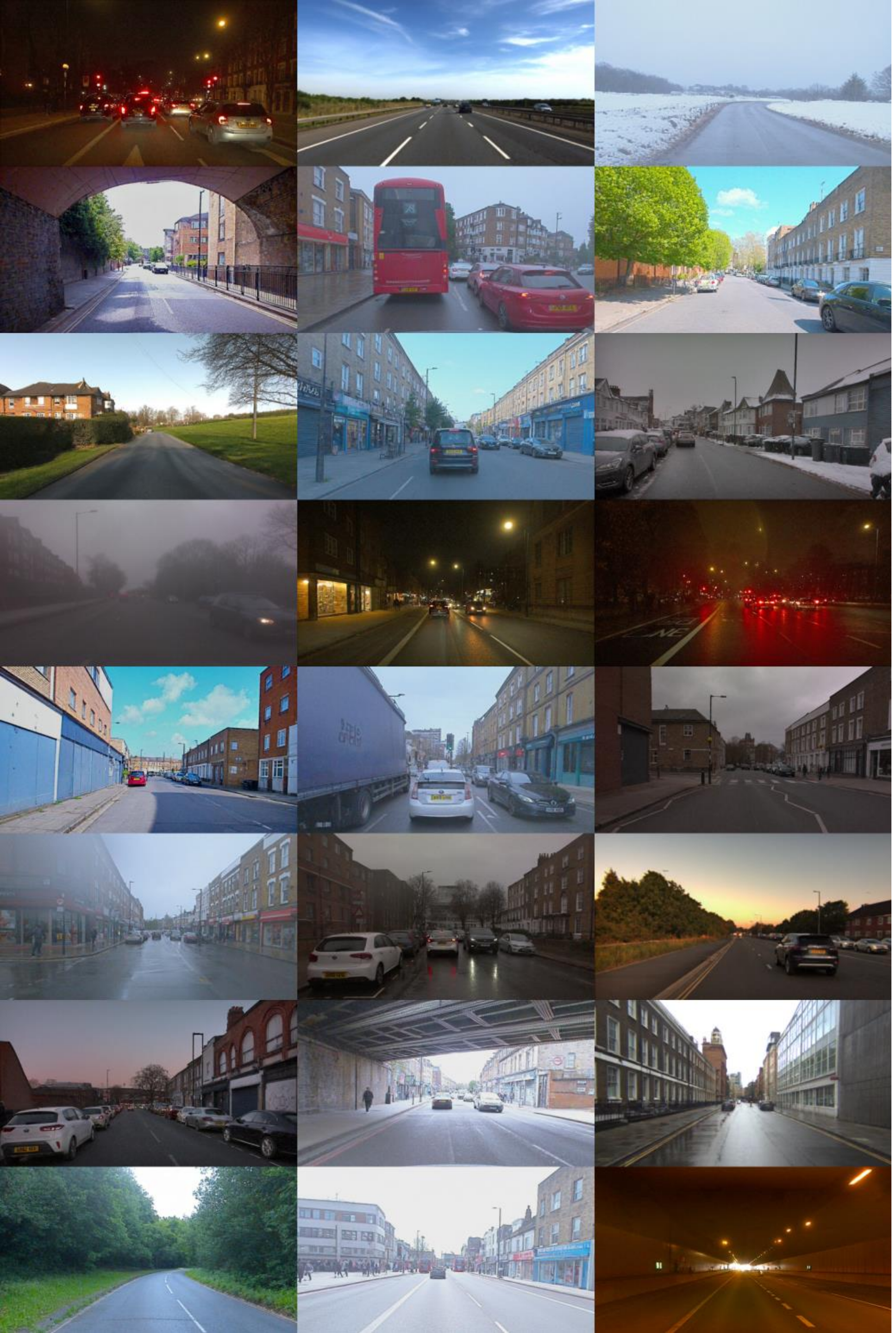
Introducing GAIA-1: A Cutting-Edge Generative AI Model for Autonomy

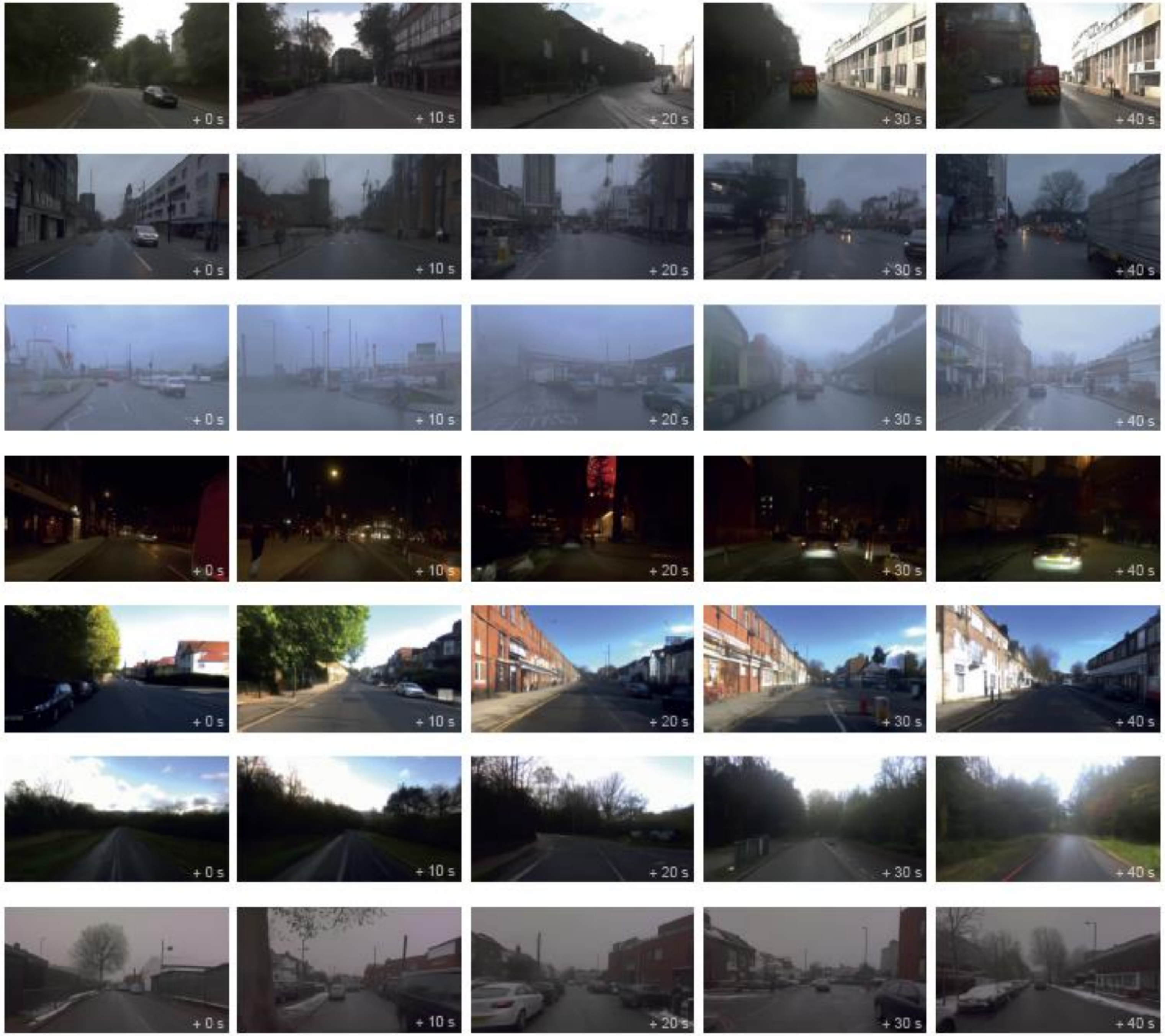
Why is GAIA-1 interesting for autonomous driving?

- *Offline*: Generating „unlimited“ video training/validation data, including some corner cases not seen in GAIA-1 training material
- *Online*: Can it even provide „a number of futures“ for better trajectory planning?

How does GAIA-1 technically work?

- GAIA-1 is a generative sequence world model for autonomous driving R&D
- Any video, text, and action prompts are individually tokenized
- After tokenization, the prompts are conditions into a recurrently excecuted world model …
- … which delivers a future image token sequence, …
- … which is input to a recurrently executed diffusion video decoder, delivering a respective video sequence from it.





[Hu, Anthony, et al. "GAIA-1: A Generative World Model for Autonomous Driving." *arXiv preprint arXiv:2309.17080* (2023)]

GAIA-1

New possibilities in autonomous driving R&D

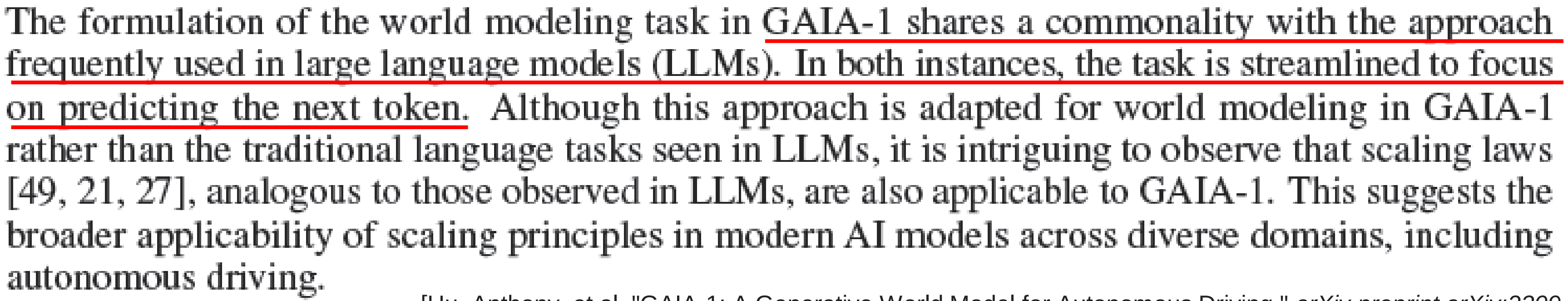
https://youtu.be/5Jx2QgEUZUI

Overview

From speech to language to vision

GAIA-1 technology observations:

- Text tokenizer, world model, and video decoder are … transformer encoder and/or decoder models
  … executed recurrently to produce output step-by-step
- The very same recurrent execution of transformer models is also used in …
  … end-to-end automatic speech recognition (E2E ASR): ENC/DEC transformer, and in …
  … large language models (LLMs, e.g., ChatGPT): DEC transformer



The formulation of the world modeling task in GAIA-1 shares a commonality with the approach frequently used in large language models (LLMs). In both instances, the task is streamlined to focus on predicting the next token. Although this approach is adapted for world modeling in GAIA-1 rather than the traditional language tasks seen in LLMs, it is intriguing to observe that scaling laws [49, 21, 27], analogous to those observed in LLMs, are also applicable to GAIA-1. This suggests the broader applicability of scaling principles in modern AI models across diverse domains, including autonomous driving. *

[Hu, Anthony, et al. "GAIA-1: A Generative World Model for Autonomous Driving." *arXiv preprint arXiv:2309.17080* (2023)]

⇒ Idea of the talk:

Let‘s explore speech and language tech first, namely:

(Section 1) E2E ASR

(Section 2) LLMs

(Section 3) GAIA-1 (finally, knowing transformers already in depth)

* Not wrong, but misleading!
While LLMs have same input and output tokens, GAIA-1 world model doesn‘t: The input is a multimodal token, thereby asking for an ENC/DEC transformer model (as in E2E ASR)

1. End-to-End Automatic Speech Recognition (E2E ASR)

Attention-based encoder-decoder (AED) models

Autoregressive decoding of output sequence tokens, token-by-token …

Token, here: ID for letters/digits/signs (~40), words (~300000),
or for so-called byte-pair encodings (subwords) (30000…50000)

The *entire* feature sequence ($\mathbf{x}_1^T$, from $1 \dots T$) is first encoded into a *hidden* representation sequence $\mathbf{h}_1^T$ of same length
$\Rightarrow$ Streaming not possible, would require modifications

At each decoder token timestep, the decoder uses the attention function to gather relevant timesteps from the hidden representation

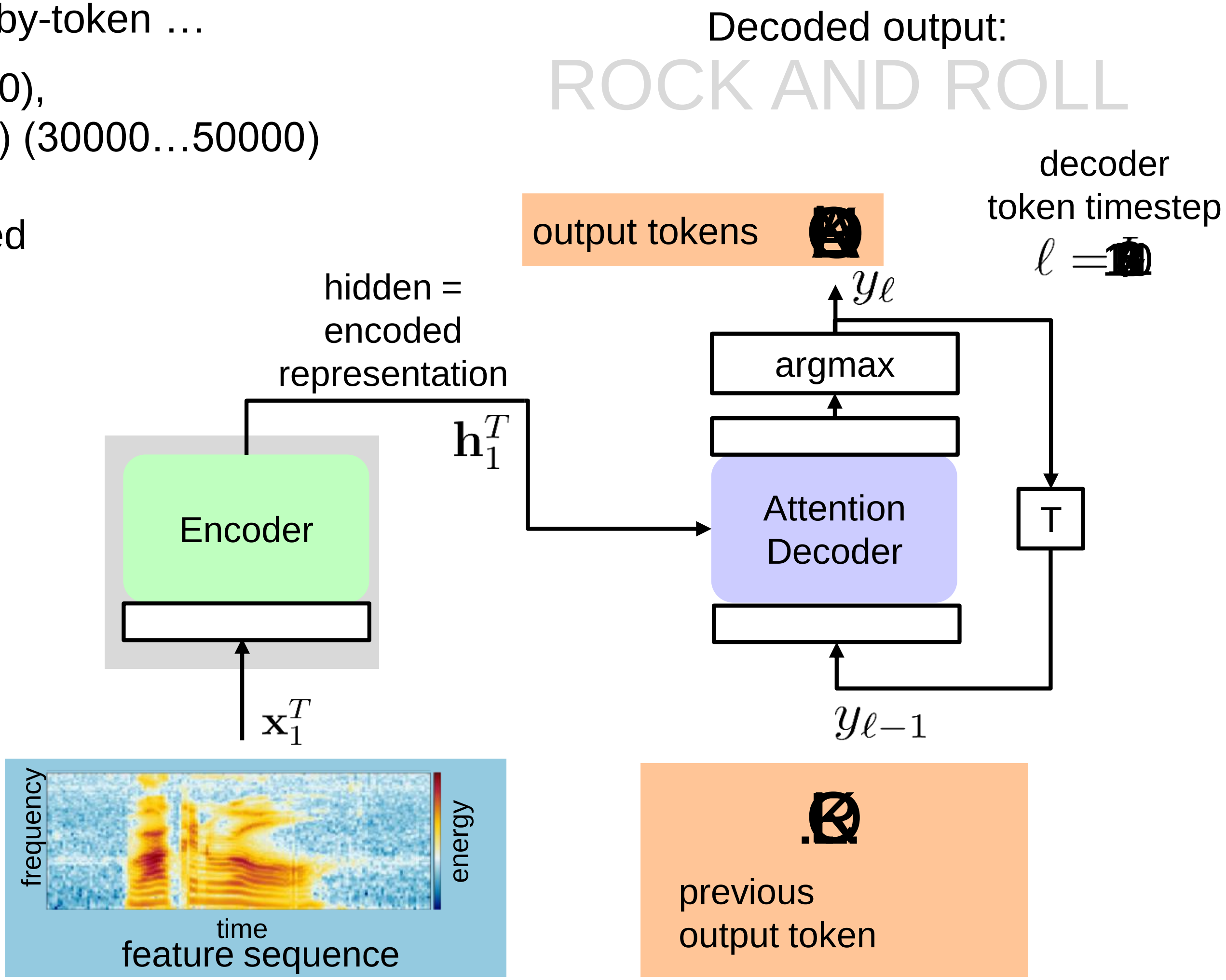
AED models perform sequence-to-sequence mapping:

$$\mathbf{x}_1^T \rightarrow \mathbf{h}_1^T \rightarrow y_1, y_2, \dots, y_\ell, \dots, y_L$$

AED models require large amounts of training data, but achieve state-of-the-art performance on several datasets

Among the common architectures is the all-attention-based transformer model

[Vaswani et al., „*Attention is All You Need*“, arXiv:1706.03762, 2017]



1. E2E ASR
The transformer AED model

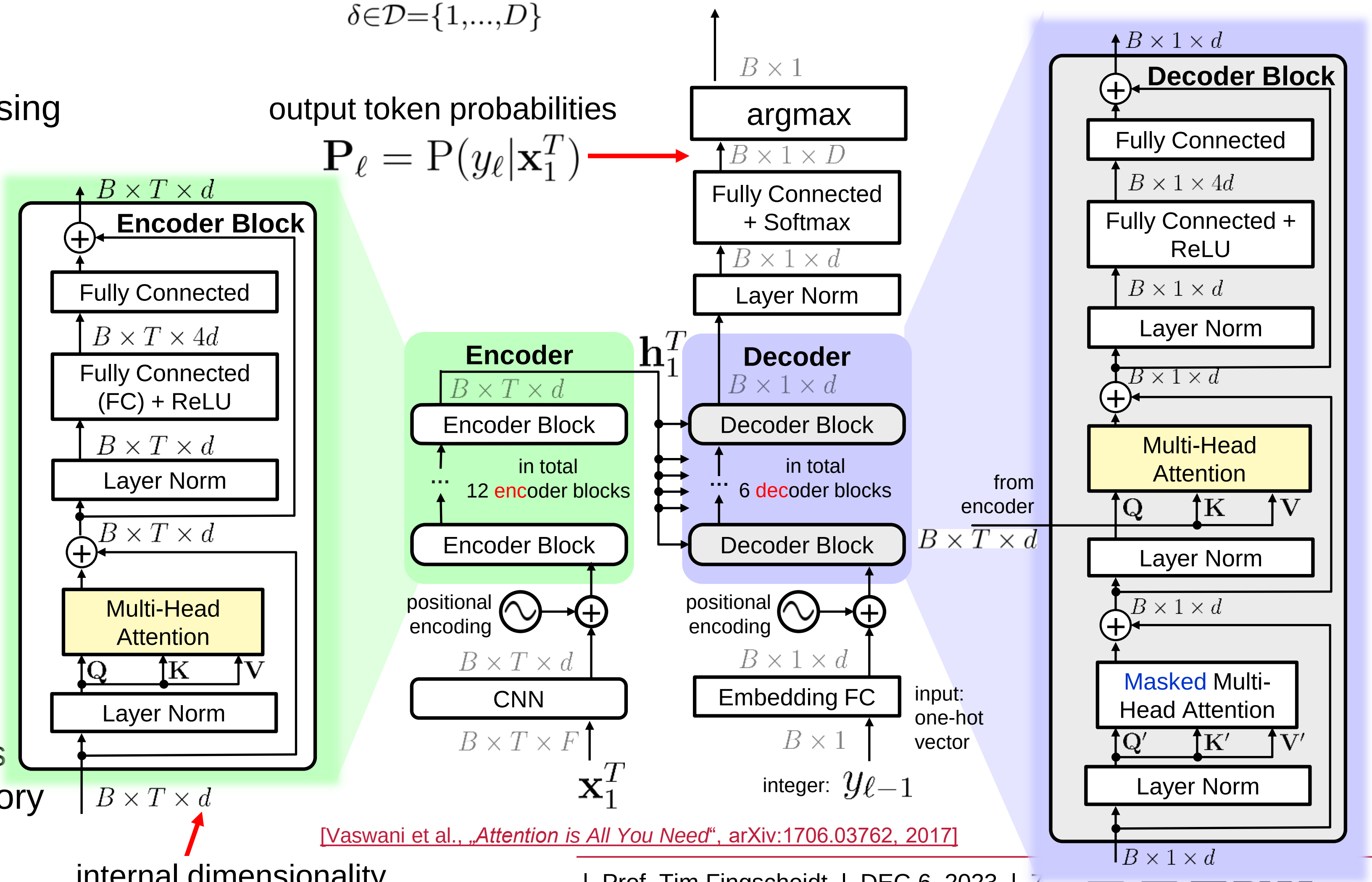
No architectural recurrency at all, bypasses used
⇒ very deep models possible

- Encoder consists of encoder blocks using (linear) self-attention and (non-linear) FC layers with residual bypasses
- Decoder in addition uses cross attention (also called: encoder-decoder attention)
- Positional information is lost in the attention layers, therefore, position needs to be encoded both in the encoder and decoder input
- Masked MHA:
  In training, future time steps are masked to zero. In inference, previous timesteps are read from internal memory

$$\underset{\delta\in\mathcal{D}=\{1,\ldots,D\}}{\arg\max}\ (\mathrm{P}_{\ell,\delta}) = y_\ell$$

output token probabilities

$$\mathbf{P}_\ell = \mathrm{P}(y_\ell|\mathbf{x}_1^T)$$



internal dimensionality
(not dependent on $\dim(\mathbf{x}_t)$)

[Vaswani et al., „*Attention is All You Need*“, arXiv:1706.03762, 2017]

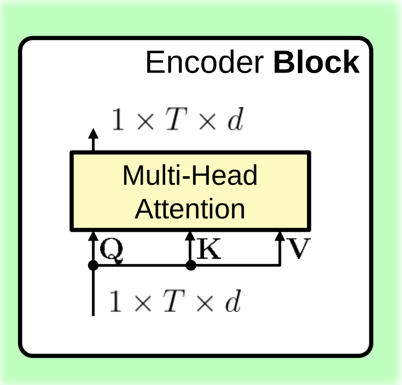
1. E2E ASR

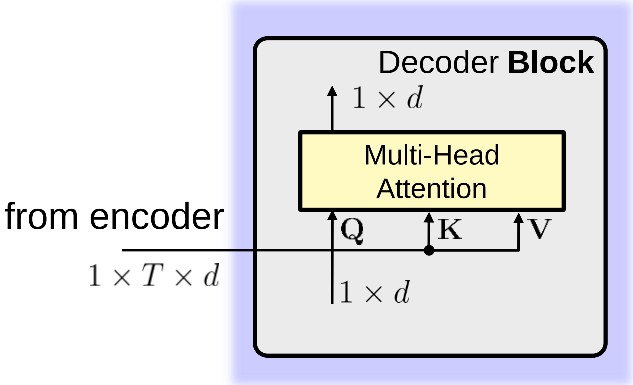
MHA function – Self-attention & cross attention

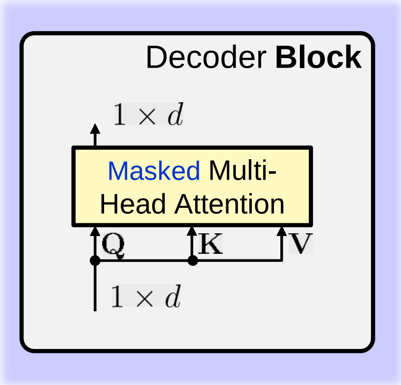
Encoder self-attention:
„Which encoder frame timesteps t (input) relate to which other encoder timesteps t (input) relevantly?“

Encoder-**decoder cross** attention:
„Which encoder frame timesteps t (input) are relevant for the current decoder token timestep ℓ (output)?“

Decoder masked MHA (**self**-attention):
„Which already decoded token timesteps $1,\ldots,\ell-1$ (previous outputs) are relevant for the current decoder token timestep ℓ (output)?“







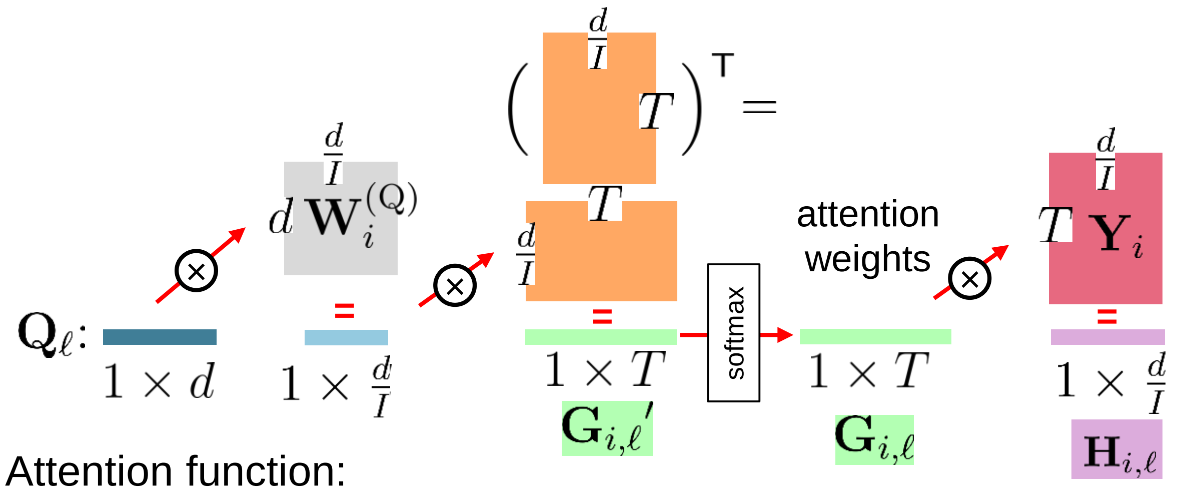
d : dimension of internal representation both of input features and of tokens

1. E2E ASR

Multi-head attention (MHA) function, inference setup

Encoder-decoder (= cross) attention:

„Which encoder frame timesteps t (input) are relevant for the current decoder token timestep ℓ (output)?“

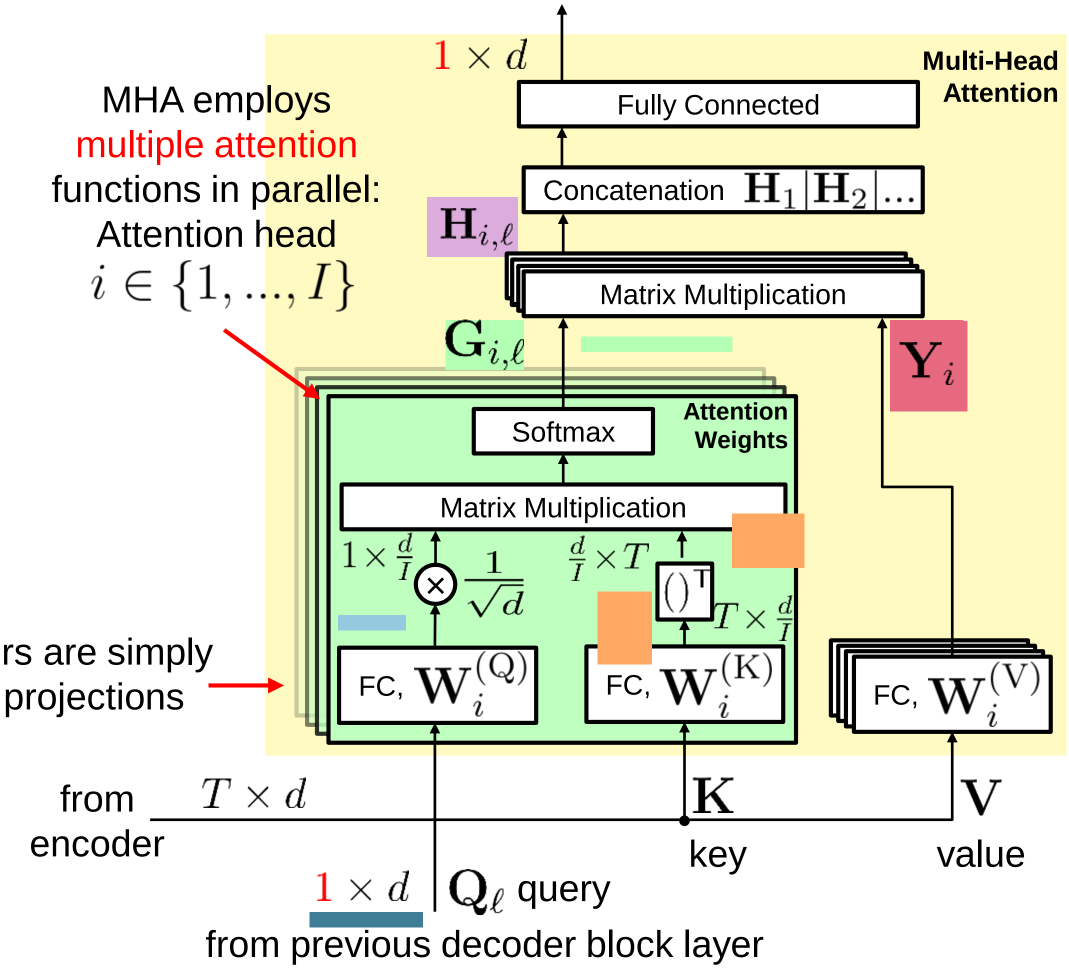


Attention function:

$$\mathbf{H}_{i,\ell} = \underbrace{\mathbf{softmax}\left(\frac{\mathbf{Q}_\ell \mathbf{W}_i^{(\mathrm{Q})} \left(\mathbf{K}\mathbf{W}_i^{(\mathrm{K})}\right)^{\mathrm{T}}}{\sqrt{d}}\right)}_{\text{attention weights } \mathbf{G}_{i,\ell}} \underbrace{\mathbf{V}\mathbf{W}_i^{(\mathrm{V})}}_{\text{value projections } \mathbf{Y}_i}$$

head index

Attention weights sum up to one over all T input encoder indices: $\sum_t G_{i,\ell,t} = 1$

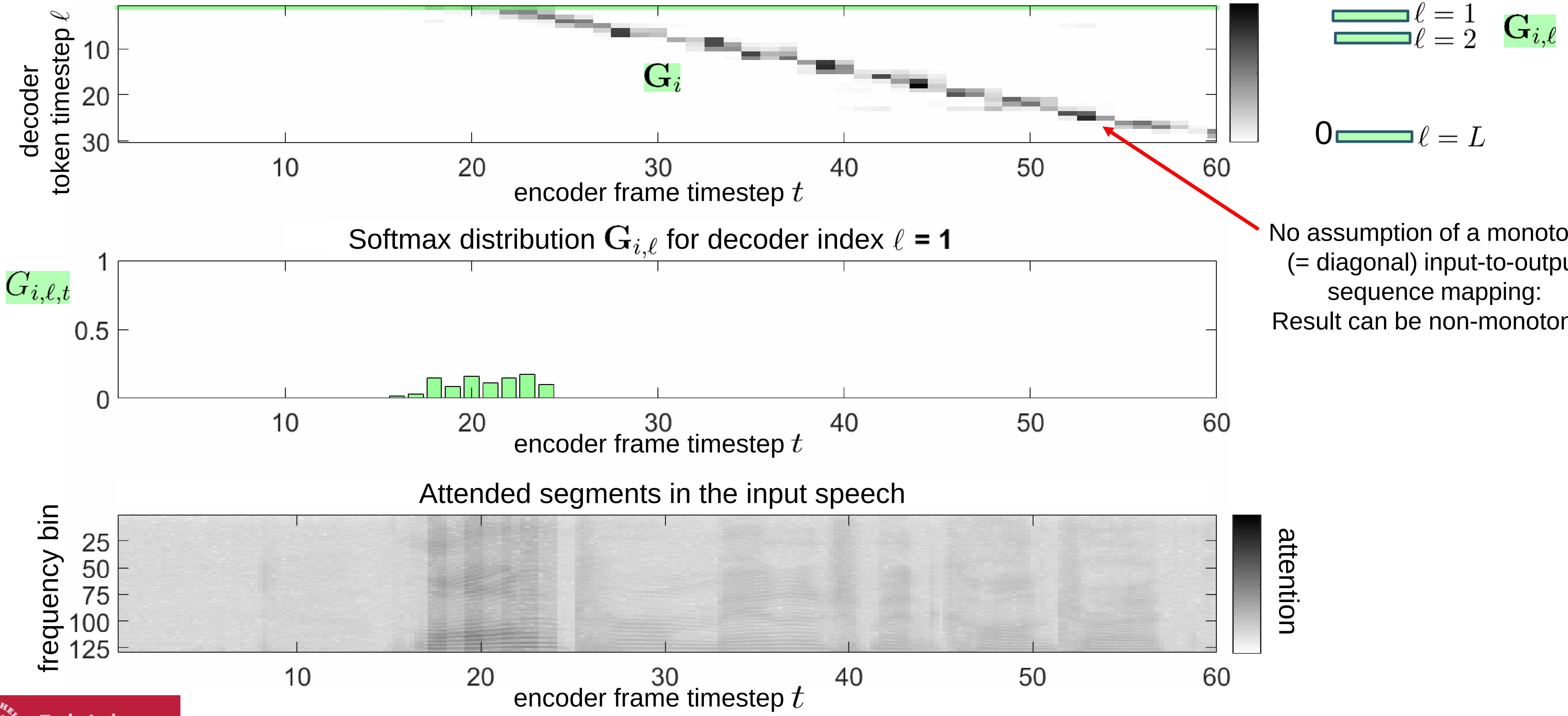


MHA employs multiple attention functions in parallel: Attention head $i \in \{1, \ldots, I\}$

FC layers are simply linear projections

1. E2E ASR

The attention mechanism



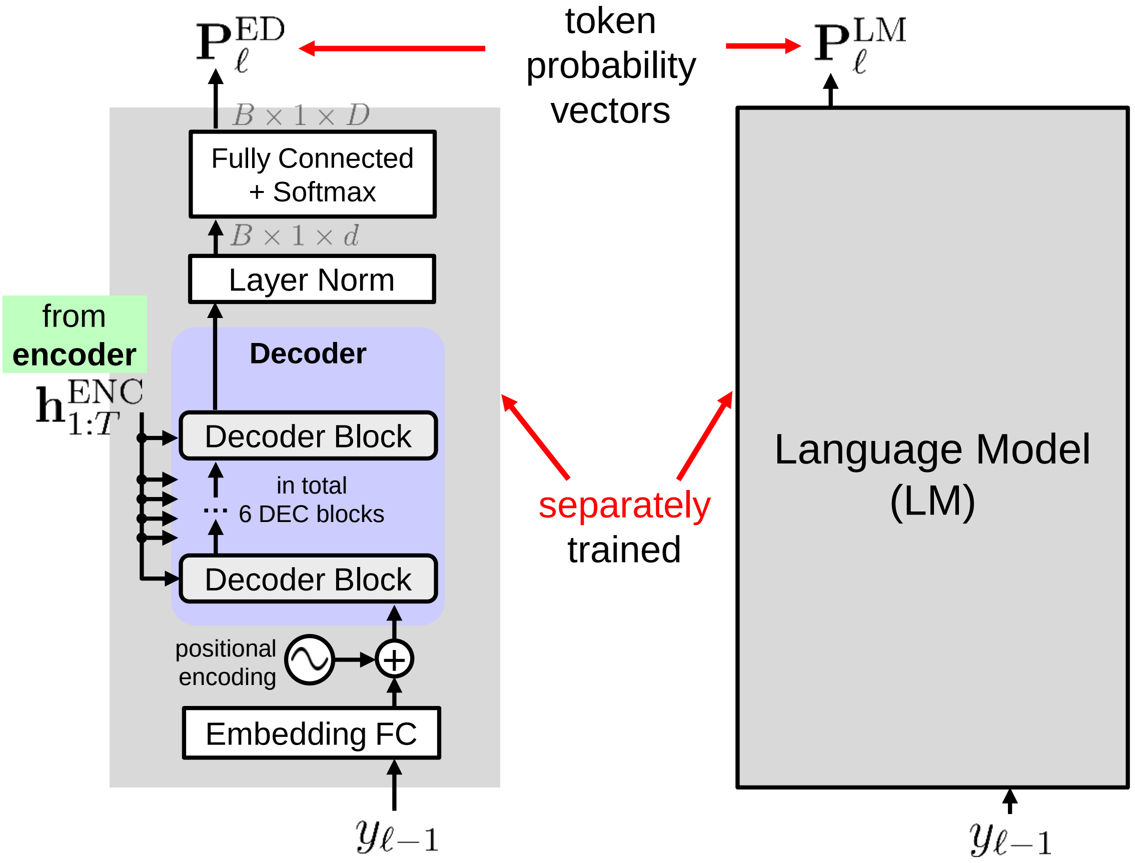
Softmax distribution $\mathbf{G}_{i,\ell}$ for decoder index ℓ **= 1**

Attended segments in the input speech

No assumption of a monotonic (= diagonal) input-to-output sequence mapping:
Result can be non-monotonic!

1. E2E ASR

… with external language model (LM)



The encoder-decoder ASR transformer model is trained on paired data (audio and text)

The language model is trained on huge amounts of unpaired data (text only)

Shallow fusion = Use of a LM in the context of E2E ASR:

For each decoding time step ℓ :
Combination of output probabilities during inference:

$$\log \mathbf{P}_\ell = \log \mathbf{P}_\ell^{\mathrm{ED}} + \lambda \log \mathbf{P}_\ell^{\mathrm{LM}}$$

~posterior likelihood prior

Final output token decision:

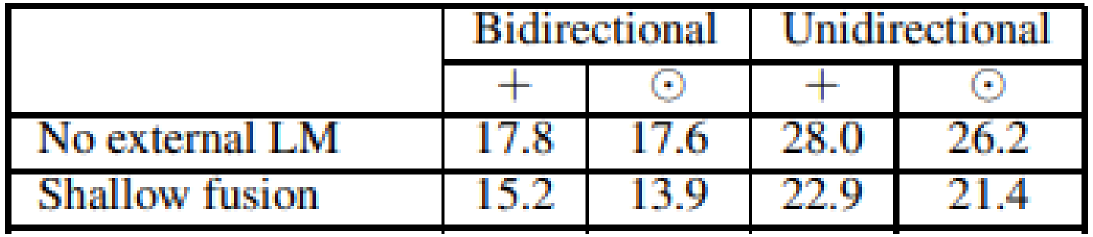
$$y_\ell = \underset{\delta\in\mathcal{D}=\{1,\ldots,D\}}{\arg\max} \; (\mathbf{P}_{\ell,\delta})$$

output token

probs

The language model weight λ balances the LM contribution

Word error rates with or without external LM:



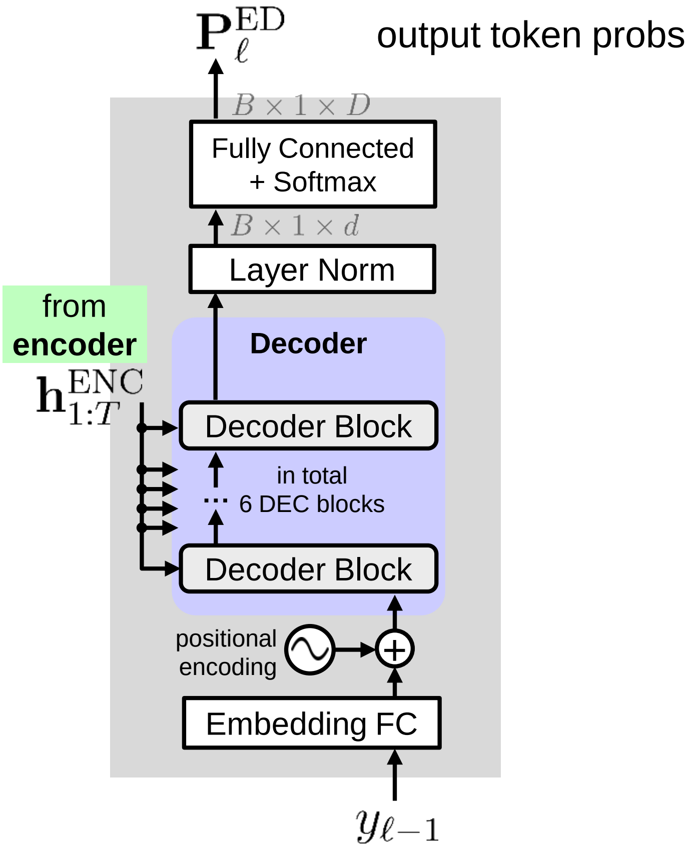
	Bidirectional		Unidirectional	
	+	⊙	+	⊙
No external LM	17.8	17.6	28.0	26.2
Shallow fusion	15.2	13.9	22.9	21.4

[G. Gulcehre et al., *“On Using Monolingual Corpora in Neural Machine Translation”*, 2015]

[E. McDermott et al., *“A Density Ratio Approach to Language Model Fusion in End-to-End Automatic Speech Recognition”*, Proc. of ASRU, 2019]

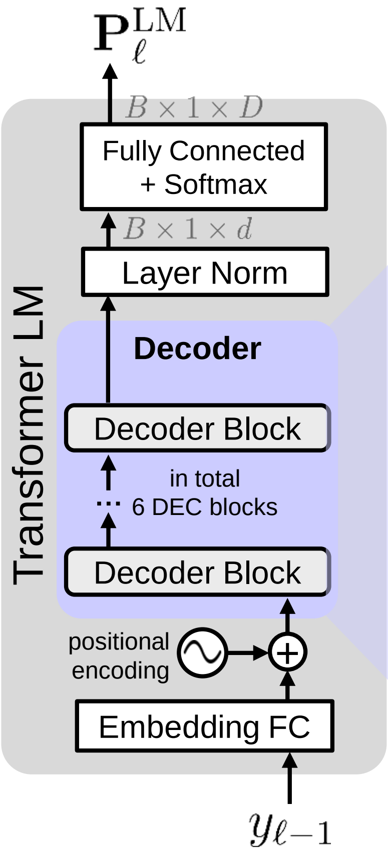
2. **Language Model (LM)**
A decoder-only model

From now on, in Section 2, we look at LMs only!
And we look on their usefulness standalone!

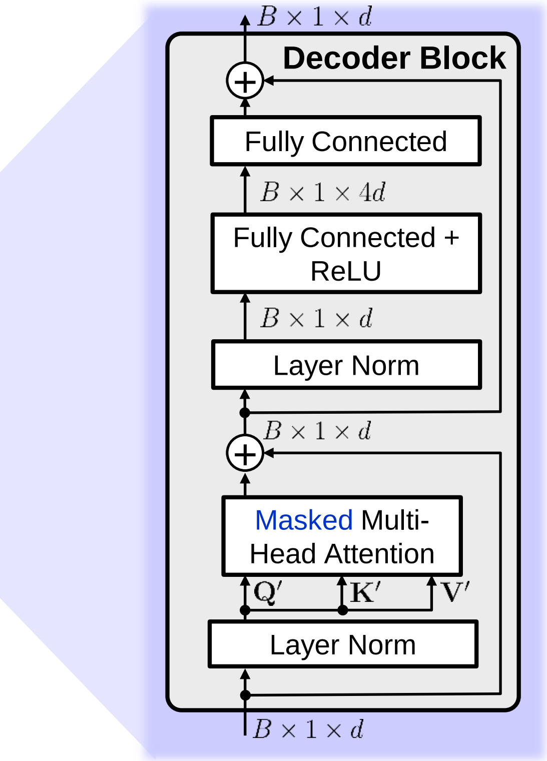


Recap: Original transformer encoder-decoder ASR model [1]

[1] Vaswani et al., „Attention is All You Need“, arXiv:1706.03762, 2017



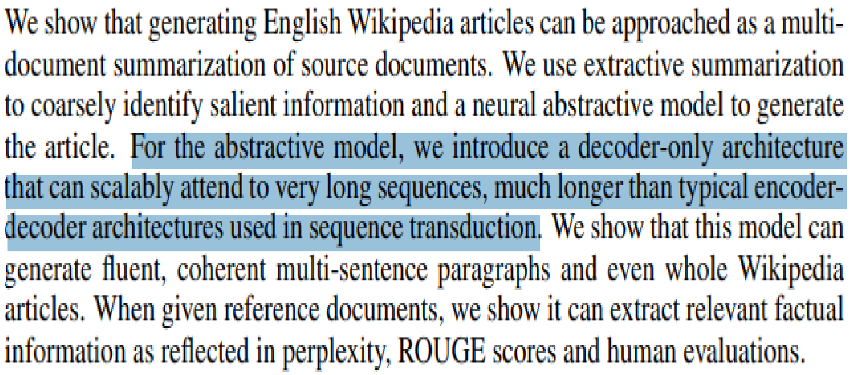
Here: Transformer language model [2] using decoder blocks



Transformer decoder block in decoder-only transformer language model [2]

Removed:
encoder-decoder
multi-head
cross attention!

Kept:
multi-head
self-attention



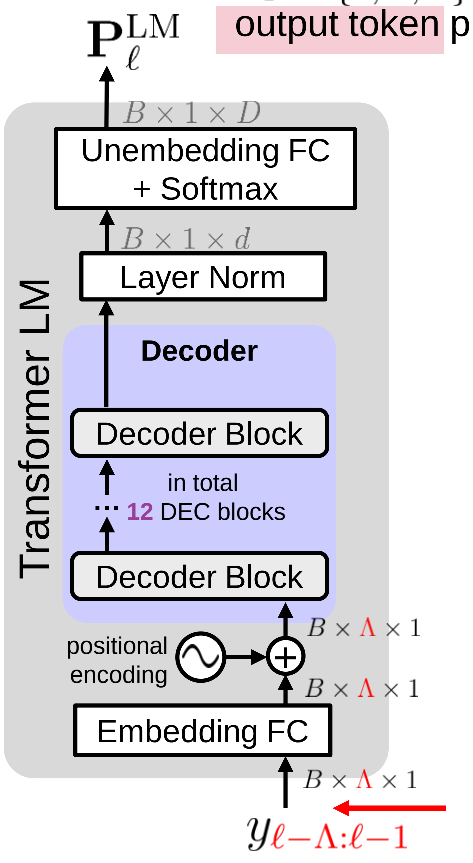
We show that generating English Wikipedia articles can be approached as a multi-document summarization of source documents. We use extractive summarization to coarsely identify salient information and a neural abstractive model to generate the article. For the abstractive model, we introduce a decoder-only architecture that can scalably attend to very long sequences, much longer than typical encoder-decoder architectures used in sequence transduction. We show that this model can generate fluent, coherent multi-sentence paragraphs and even whole Wikipedia articles. When given reference documents, we show it can extract relevant factual information as reflected in perplexity, ROUGE scores and human evaluations.

[2] Liu et al., „Generating Wikipedia by Summarizing Long Sequences“. ICLR, 2018

2. Large Language Model (LLM)

Generative pre-trained transformer (GPT)

$$y_\ell = \underset{\delta \in \mathcal{D}=\{1,\ldots,D\}}{\arg\max} \left(\mathrm{P}^{\mathrm{LM}}_{\ell,\delta}\right)$$



During pre-training, the previous predictions with a context window of length Λ are used to predict the next token probabilities $\mathbf{P}^{\mathrm{LM}}_\ell$

Generative pre-trained transformer (GPT) language model [1]:

Unsupervised pre-training by next token prediction:

Loss function: $J = \sum_{\ell \in \mathcal{L}} J_\ell = -\sum_{\ell \in \mathcal{L}} \log \mathrm{P}^{\mathrm{LM}}_{\ell,\delta=\overline{y}_\ell}$ ← token probability of correct token $\overline{y}_\ell$

with the decoding step $\ell \in \mathcal{L} = \{1, 2, \ldots, L\}$
and L is the length of the ground truth sequence $(\overline{y}_\ell) = \overline{y}_{1:L}$

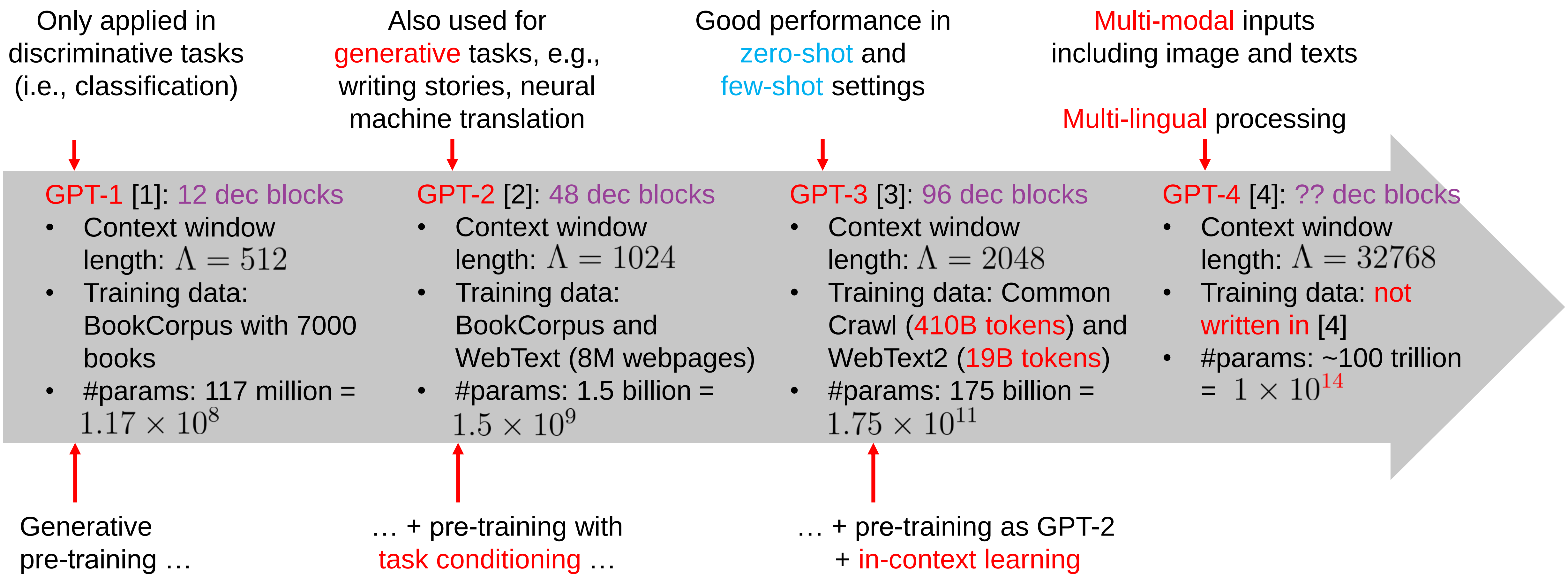
The GPT-1 [1] language model …
- … has a context window of length $\Lambda = 512$
- … uses byte-pair encoding (BPE), the vocabulary size is 40k
- … consists of 12 decoder blocks with in total 117M parameters
- … requires 1 month on 8 GPUs for pre-training ← A lot! But still university-grade…
- … is pre-trained by publicly available 7000 books
- … code is published on GitHub:

https://github.com/openai/finetune-transformer-lm

[1] Radford et al. "Improving Language Understanding by Generative Pre-Training." (2018).

2. LLM

The rise of GPT: From GPT-1 to GPT4



Only applied in discriminative tasks (i.e., classification)

Also used for generative tasks, e.g., writing stories, neural machine translation

Good performance in zero-shot and few-shot settings

Multi-modal inputs including image and texts

Multi-lingual processing

GPT-1 [1]: 12 dec blocks

- Context window length: $\Lambda = 512$
- Training data: BookCorpus with 7000 books
- #params: 117 million = 1.17×10^8

GPT-2 [2]: 48 dec blocks

- Context window length: $\Lambda = 1024$
- Training data: BookCorpus and WebText (8M webpages)
- #params: 1.5 billion = 1.5×10^9

GPT-3 [3]: 96 dec blocks

- Context window length: $\Lambda = 2048$
- Training data: Common Crawl (410B tokens) and WebText2 (19B tokens)
- #params: 175 billion = 1.75×10^{11}

GPT-4 [4]: ?? dec blocks

- Context window length: $\Lambda = 32768$
- Training data: not written in [4]
- #params: ~100 trillion = 1×10^{14}

Generative pre-training …

… + pre-training with task conditioning …

… + pre-training as GPT-2 + in-context learning

[1] Radford et al. "Improving Language Understanding By Generative Pre-training." (2018).

[2] Radford et al. "Language Models Are Unsupervised Multitask Learners." *OpenAI blog* 1.8 (2019): 9.

[3] Brown et al. "Language Models Are Few-Shot Learners." *in Proc. of NeurIPS*, virtual, Dec, 2020, 1877-1901.

[4] OpenAI "GPT-4 Technical Report." *arXiv* (2023): 2303-08774.

2. LLM

GPT-4: Multi-modal large language model

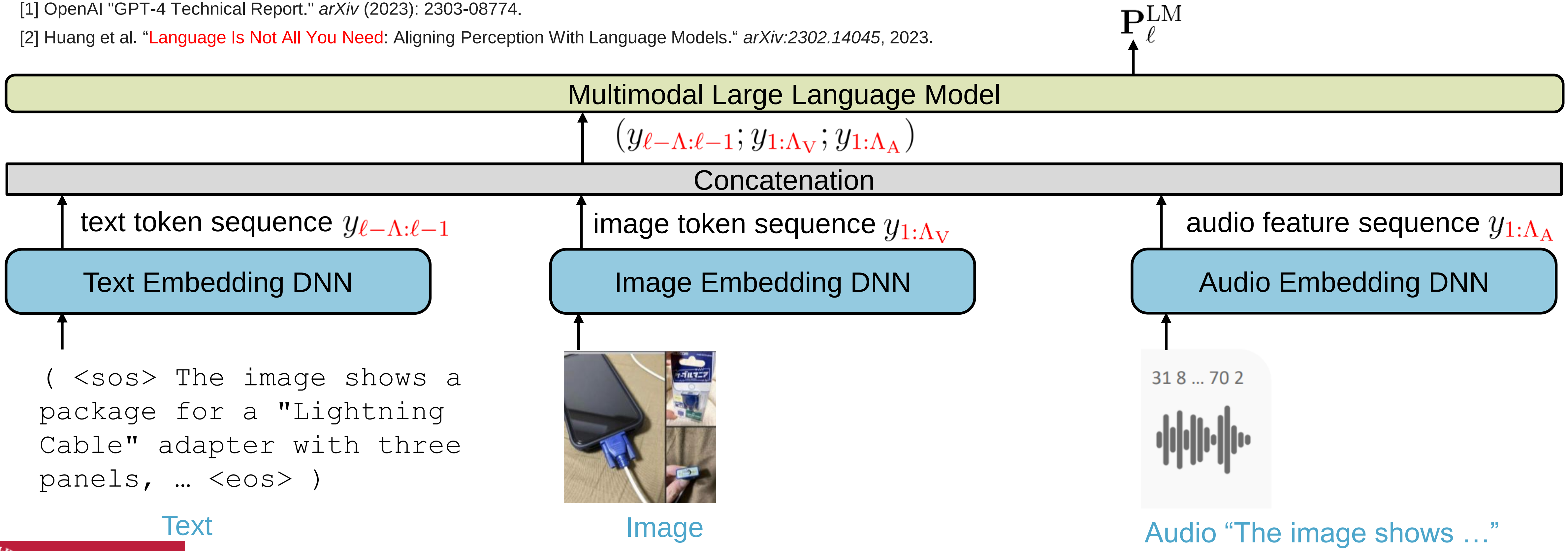
GPT4 can take both text and image as input.

The technical report from OpenAI [1] doesn’t give any details on model architecture and training.

The following technique is based on a different multi-modal large language model (LLM) [2]:

[1] OpenAI "GPT-4 Technical Report." *arXiv* (2023): 2303-08774.

[2] Huang et al. “Language Is Not All You Need: Aligning Perception With Language Models.“ *arXiv:2302.14045*, 2023.



$\mathbf{P}_{\ell}^{\mathrm{LM}}$

Multimodal Large Language Model

$\left(y_{\ell-\Lambda:\ell-1}; y_{1:\Lambda_{\mathrm{V}}}; y_{1:\Lambda_{\mathrm{A}}}\right)$

Concatenation

text token sequence $y_{\ell-\Lambda:\ell-1}$

image token sequence $y_{1:\Lambda_{\mathrm{V}}}$

audio feature sequence $y_{1:\Lambda_{\mathrm{A}}}$

Text Embedding DNN

Image Embedding DNN

Audio Embedding DNN

```
( <sos> The image shows a
package for a "Lightning
Cable" adapter with three
panels, … <eos> )
```

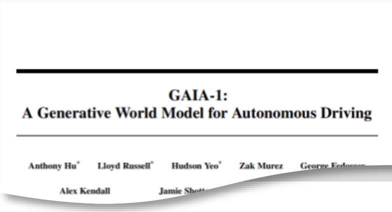
Text

Image

Audio “The image shows …”

3. Overview of GAIA-1 (Inference)

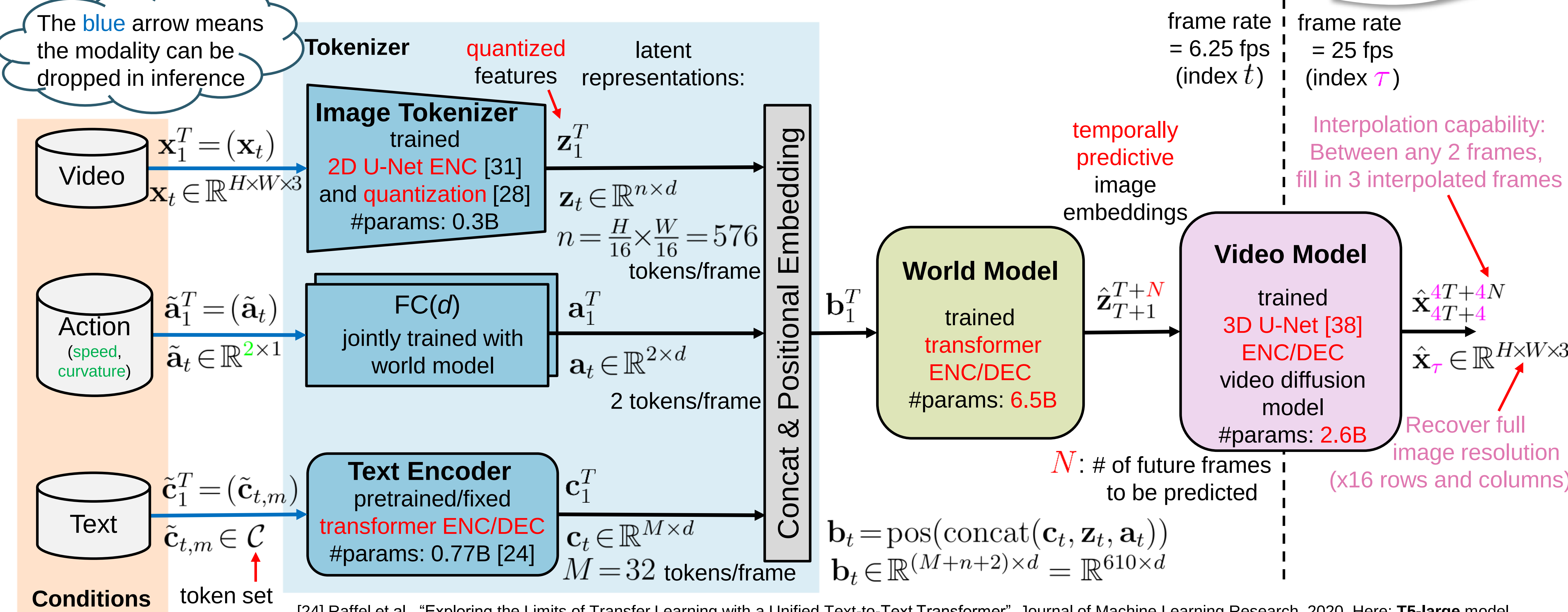
[1] Hu, Anthony, et al. "GAIA-1: A Generative World Model for Autonomous Driving." *arXiv preprint arXiv:2309.17080* (2023).



GAIA-1:
A Generative World Model for Autonomous Driving

Anthony Hu* Lloyd Russell* Hudson Yeo* Zak Murez George Fedoseev

Alex Kendall Jamie Shott



The blue arrow means the modality can be dropped in inference

Conditions

Video

$\mathbf{x}_1^T = (\mathbf{x}_t)$

$\mathbf{x}_t \in \mathbb{R}^{H \times W \times 3}$

Action (speed, curvature)

$\tilde{\mathbf{a}}_1^T = (\tilde{\mathbf{a}}_t)$

$\tilde{\mathbf{a}}_t \in \mathbb{R}^{2 \times 1}$

Text

$\tilde{\mathbf{c}}_1^T = (\tilde{\mathbf{c}}_{t,m})$

$\tilde{\mathbf{c}}_{t,m} \in \mathcal{C}$

token set

Tokenizer

quantized features

latent representations:

Image Tokenizer
trained
2D U-Net ENC [31]
and quantization [28]
#params: 0.3B

$\mathbf{z}_1^T$

$\mathbf{z}_t \in \mathbb{R}^{n \times d}$

$n = \frac{H}{16} \times \frac{W}{16} = 576$ tokens/frame

FC(*d*)
jointly trained with world model

$\mathbf{a}_1^T$

$\mathbf{a}_t \in \mathbb{R}^{2 \times d}$

2 tokens/frame

Text Encoder
pretrained/fixed
transformer ENC/DEC
#params: 0.77B [24]

$\mathbf{c}_1^T$

$\mathbf{c}_t \in \mathbb{R}^{M \times d}$

$M = 32$ tokens/frame

Concat & Positional Embedding

$\mathbf{b}_1^T$

$\mathbf{b}_t = \text{pos}(\text{concat}(\mathbf{c}_t, \mathbf{z}_t, \mathbf{a}_t))$

$\mathbf{b}_t \in \mathbb{R}^{(M+n+2) \times d} = \mathbb{R}^{610 \times d}$

World Model
trained
transformer
ENC/DEC
#params: 6.5B

temporally predictive image embeddings

$\hat{\mathbf{z}}_{T+1}^{T+N}$

N: # of future frames to be predicted

frame rate = 6.25 fps (index t)

frame rate = 25 fps (index τ)

Video Model
trained
3D U-Net [38]
ENC/DEC
video diffusion model
#params: 2.6B

Interpolation capability: Between any 2 frames, fill in 3 interpolated frames

$\hat{\mathbf{x}}_{4T+4}^{4T+4N}$

$\hat{\mathbf{x}}_\tau \in \mathbb{R}^{H \times W \times 3}$

Recover full image resolution (x16 rows and columns)

[24] Raffel et al., "Exploring the Limits of Transfer Learning with a Unified Text-to-Text Transformer". Journal of Machine Learning Research, 2020. Here: **T5-large** model.
[28] Oord et al., „Neural Discrete Representation Learning". In Proc. of NeurIPS, 2017.
[31] Ronneberger et al. „U-Net: Convolutional Networks for Biomedical Image Segmentation". In Medical Image Computing and Computer-Assisted Intervention (MICCAI), 2015.
[38] J. Ho, T. Salimans, A. Gritsenko, W. Chan, M. Norouzi, and D. J. Fleet, "Video Diffusion Models." arXiv, Jun. 22, 2022.

3. GAIA-1 Tokenizer

Image tokenizer (training)

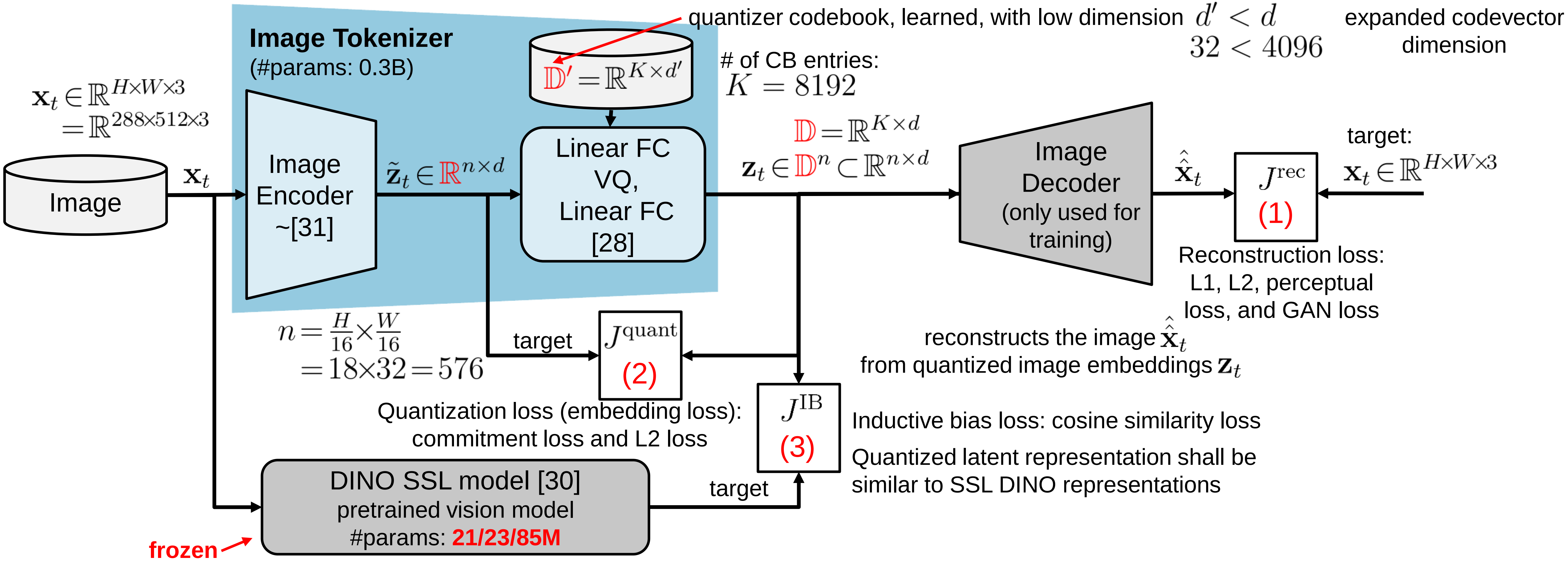


Image Tokenizer

(#params: 0.3B)

quantizer codebook, learned, with low dimension $d' < d$

$32 < 4096$

expanded codevector dimension

$\mathbb{D}' = \mathbb{R}^{K \times d'}$

# of CB entries: $K = 8192$

$\mathbf{x}_t \in \mathbb{R}^{H \times W \times 3} = \mathbb{R}^{288 \times 512 \times 3}$

Image

$\mathbf{x}_t$

Image Encoder ~[31]

$\tilde{\mathbf{z}}_t \in \mathbb{R}^{n \times d}$

Linear FC VQ, Linear FC [28]

$\mathbb{D} = \mathbb{R}^{K \times d}$

$\mathbf{z}_t \in \mathbb{D}^n \subset \mathbb{R}^{n \times d}$

Image Decoder (only used for training)

$\hat{\hat{\mathbf{x}}}_t$

J^{rec} (1)

target: $\mathbf{x}_t \in \mathbb{R}^{H \times W \times 3}$

Reconstruction loss: L1, L2, perceptual loss, and GAN loss

reconstructs the image $\hat{\hat{\mathbf{x}}}_t$ from quantized image embeddings $\mathbf{z}_t$

$n = \frac{H}{16} \times \frac{W}{16} = 18 \times 32 = 576$

target

J^{quant} (2)

Quantization loss (embedding loss): commitment loss and L2 loss

J^{IB} (3)

Inductive bias loss: cosine similarity loss

Quantized latent representation shall be similar to SSL DINO representations

DINO SSL model [30] pretrained vision model #params: **21/23/85M**

target

frozen

The final loss has 3 components: (1) image reconstruction loss, (2) quantization loss, (3) inductive bias loss

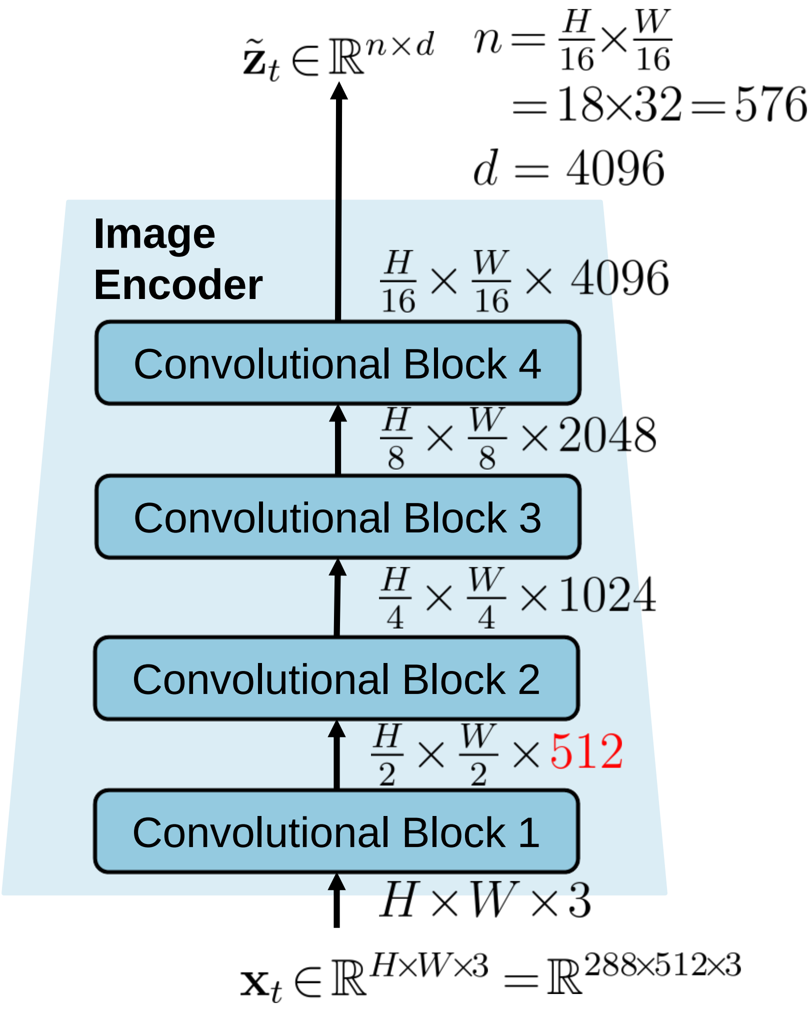
[28] Oord et al., „Neural Discrete Representation Learning". In Proc. of NeurIPS, 2017.

[31] Ronneberger et al. „U-Net: Convolutional Networks for Biomedical Image Segmentation". In Medical Image Computing and Computer-Assisted Intervention (MICCAI), 2015.

[30] Caron et al., „Emerging Properties in Self-Supervised Vision Transformers". In Proc. of ICCV, 2021.

3. GAIA-1 Tokenizer

Image encoder



$\tilde{\mathbf{z}}_t \in \mathbb{R}^{n \times d}$ $\quad n = \frac{H}{16} \times \frac{W}{16} = 18 \times 32 = 576$

$d = 4096$

Image Encoder

$\frac{H}{16} \times \frac{W}{16} \times 4096$

Convolutional Block 4

$\frac{H}{8} \times \frac{W}{8} \times 2048$

Convolutional Block 3

$\frac{H}{4} \times \frac{W}{4} \times 1024$

Convolutional Block 2

$\frac{H}{2} \times \frac{W}{2} \times 512$

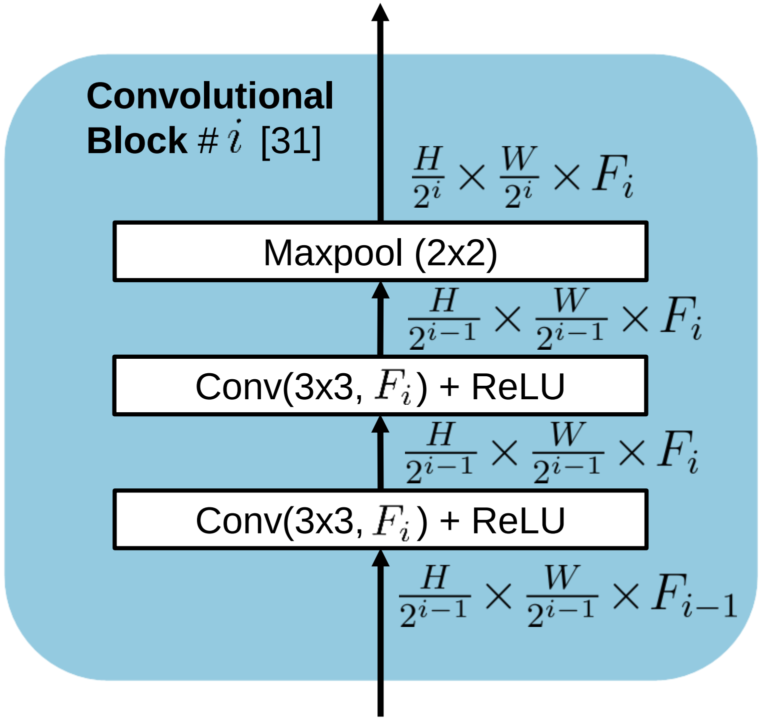
Convolutional Block 1

$H \times W \times 3$

$\mathbf{x}_t \in \mathbb{R}^{H \times W \times 3} = \mathbb{R}^{288 \times 512 \times 3}$

„The discrete autoencoder is a fully convolutional U-Net structure [31]"

- However, no architecture details to the image encoder in GAIA-1
- Here: Reverse engineering: The output dimension of the 1st convolutional block can be changed from 64 to 512 to match the 0.3B parameters written in GAIA-1



Convolutional Block # i [31]

$\frac{H}{2^i} \times \frac{W}{2^i} \times F_i$

Maxpool (2x2)

$\frac{H}{2^{i-1}} \times \frac{W}{2^{i-1}} \times F_i$

Conv(3x3, F_i) + ReLU

$\frac{H}{2^{i-1}} \times \frac{W}{2^{i-1}} \times F_i$

Conv(3x3, F_i) + ReLU

$\frac{H}{2^{i-1}} \times \frac{W}{2^{i-1}} \times F_{i-1}$

In each convolutional block [31] do:

- Upsample the feature dimension by 2 (except the 1st one) $F_i = 2F_{i-1}$
- Downsample each spatial dimension by 2

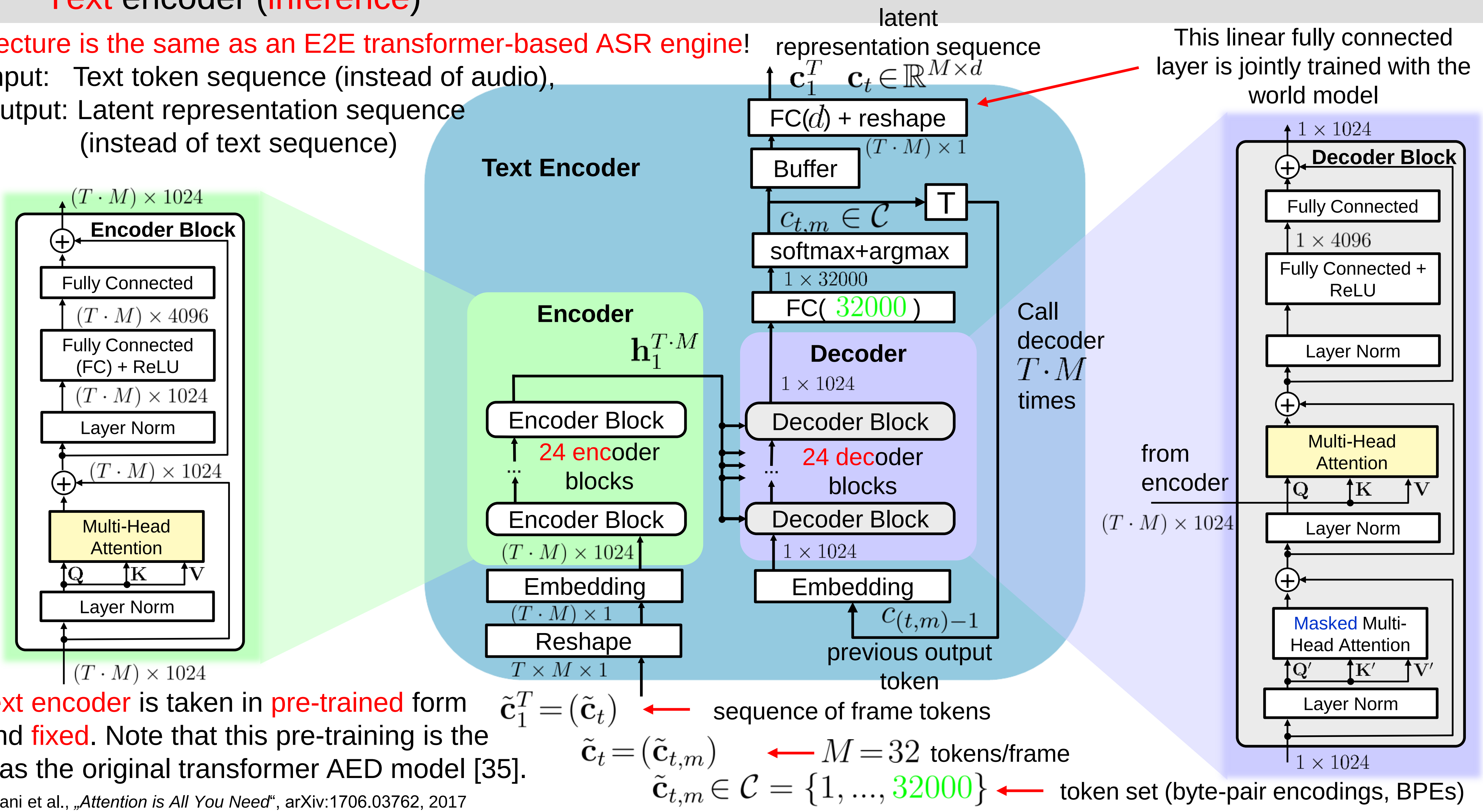
[31] Ronneberger et al. „U-Net: Convolutional Networks for Biomedical Image Segmentation". In Medical Image Computing and Computer-Assisted Intervention (MICCAI), 2015.

3. GAIA-1 Tokenizer

Text encoder (inference)

Architecture is the same as an E2E transformer-based ASR engine!

But: Input: Text token sequence (instead of audio),
Output: Latent representation sequence
(instead of text sequence)


latent
representation sequence
$\mathbf{c}_1^T \quad \mathbf{c}_t \in \mathbb{R}^{M \times d}$
This linear fully connected layer is jointly trained with the world model
Text Encoder
FC(d) + reshape
$(T \cdot M) \times 1$
Buffer
T
$c_{t,m} \in \mathcal{C}$
softmax+argmax
1×32000
FC(32000)
Call decoder $T \cdot M$ times
Encoder
$\mathbf{h}_1^{T \cdot M}$
Encoder Block
24 encoder blocks
Encoder Block
$(T \cdot M) \times 1024$
Embedding
$(T \cdot M) \times 1$
Reshape
$T \times M \times 1$
Decoder
1×1024
Decoder Block
24 decoder blocks
Decoder Block
1×1024
Embedding
$c_{(t,m)-1}$
previous output token
$\tilde{\mathbf{c}}_1^T = (\tilde{\mathbf{c}}_t)$
sequence of frame tokens
$\tilde{\mathbf{c}}_t = (\tilde{\mathbf{c}}_{t,m})$
$M = 32$ tokens/frame
$\tilde{\mathbf{c}}_{t,m} \in \mathcal{C} = \{1, \ldots, 32000\}$
token set (byte-pair encodings, BPEs)
$(T \cdot M) \times 1024$
Encoder Block
Fully Connected
$(T \cdot M) \times 4096$
Fully Connected (FC) + ReLU
$(T \cdot M) \times 1024$
Layer Norm
$(T \cdot M) \times 1024$
Multi-Head Attention
$\mathbf{Q}$ $\mathbf{K}$ $\mathbf{V}$
Layer Norm
$(T \cdot M) \times 1024$
1×1024
Decoder Block
Fully Connected
1×4096
Fully Connected + ReLU
Layer Norm
from encoder
$(T \cdot M) \times 1024$
Multi-Head Attention
$\mathbf{Q}$ $\mathbf{K}$ $\mathbf{V}$
Layer Norm
Masked Multi-Head Attention
$\mathbf{Q}'$ $\mathbf{K}'$ $\mathbf{V}'$
Layer Norm
1×1024

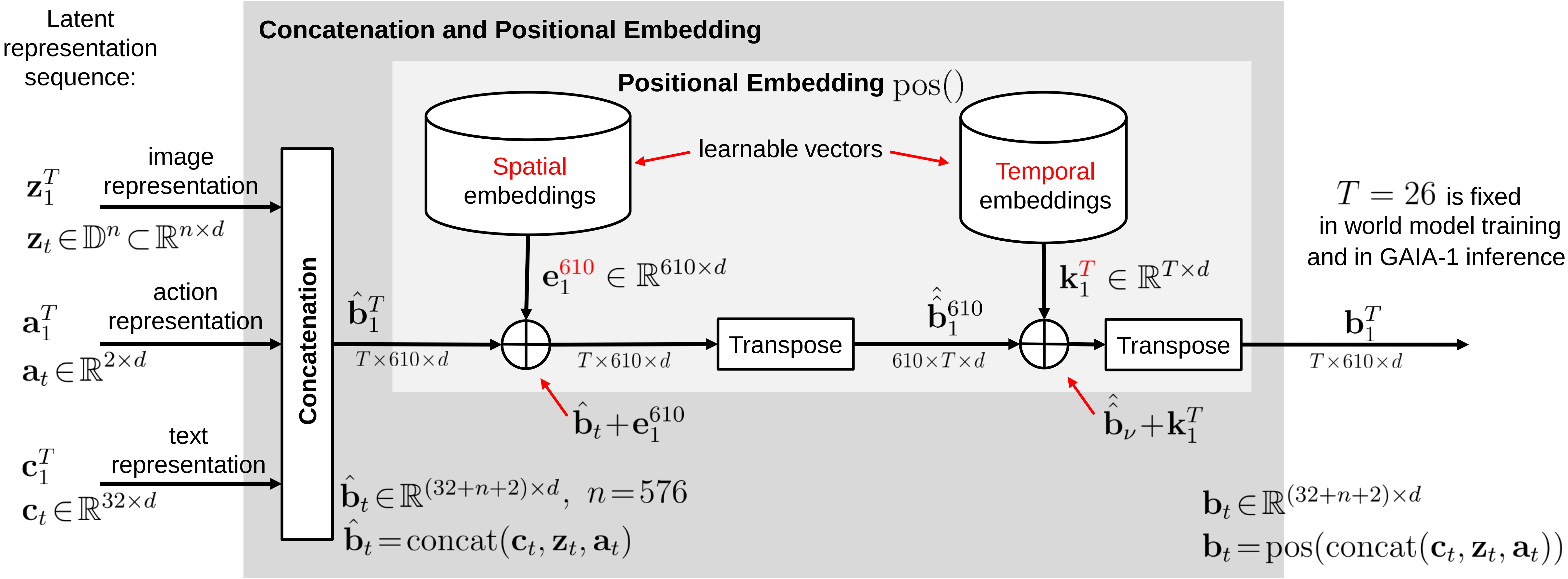
The text encoder is taken in pre-trained form [24] and fixed. Note that this pre-training is the same as the original transformer AED model [35].

[35] Vaswani et al., *„Attention is All You Need*“, arXiv:1706.03762, 2017

[24] Raffel et al., “Exploring the Limits of Transfer Learning with a Unified Text-to-Text Transformer”. Journal of Machine Learning Research, 2020. Here: **T5-large** model.

3. GAIA-1 Tokenizer

Concatenation & positional embedding



Latent representation sequence:

Concatenation and Positional Embedding

Positional Embedding $\mathrm{pos}()$

$\mathbf{z}_1^T$ image representation

$\mathbf{z}_t \in \mathbb{D}^n \subset \mathbb{R}^{n \times d}$

$\mathbf{a}_1^T$ action representation

$\mathbf{a}_t \in \mathbb{R}^{2 \times d}$

$\mathbf{c}_1^T$ text representation

$\mathbf{c}_t \in \mathbb{R}^{32 \times d}$

Concatenation

$\hat{\mathbf{b}}_1^T$

$T \times 610 \times d$

Spatial embeddings

learnable vectors

Temporal embeddings

$\mathbf{e}_1^{610} \in \mathbb{R}^{610 \times d}$

$T \times 610 \times d$

Transpose

$\hat{\hat{\mathbf{b}}}_1^{610}$

$610 \times T \times d$

$\mathbf{k}_1^T \in \mathbb{R}^{T \times d}$

Transpose

$\mathbf{b}_1^T$

$T \times 610 \times d$

$\hat{\mathbf{b}}_t + \mathbf{e}_1^{610}$

$\hat{\hat{\mathbf{b}}}_\nu + \mathbf{k}_1^T$

$T = 26$ is fixed in world model training and in GAIA-1 inference

$\hat{\mathbf{b}}_t \in \mathbb{R}^{(32+n+2) \times d}, \quad n = 576$

$\hat{\mathbf{b}}_t = \mathrm{concat}(\mathbf{c}_t, \mathbf{z}_t, \mathbf{a}_t)$

$\mathbf{b}_t \in \mathbb{R}^{(32+n+2) \times d}$

$\mathbf{b}_t = \mathrm{pos}(\mathrm{concat}(\mathbf{c}_t, \mathbf{z}_t, \mathbf{a}_t))$

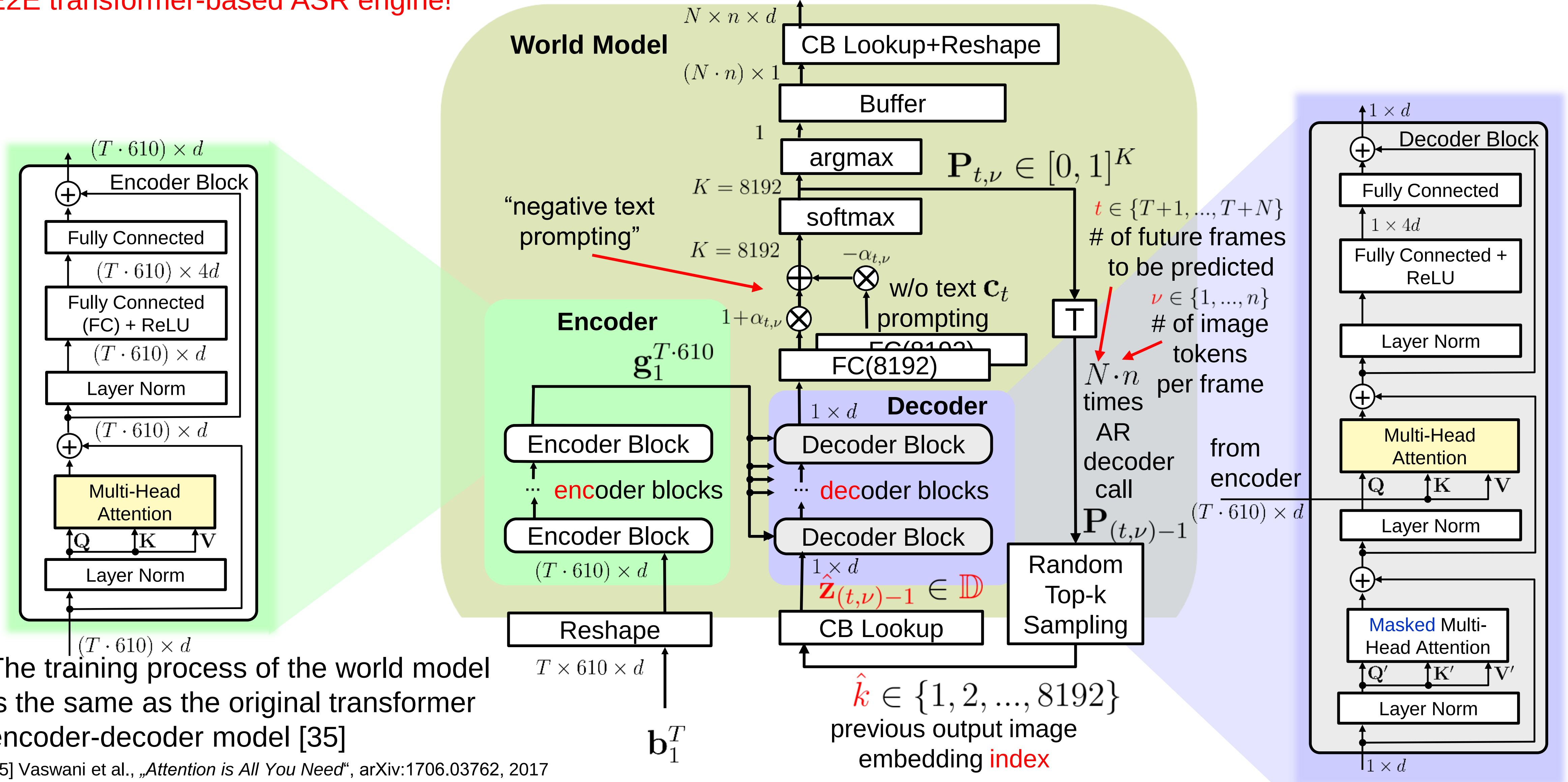
3. GAIA-1 World Model

A multimodal-in and unimodal-out system (inference)

Architecture is again similar to an E2E transformer-based ASR engine!

1 output modality: image (tokens) $\hat{\mathbf{z}}_{T+1}^{T+N}$, $\hat{\mathbf{z}}_t = (\hat{\mathbf{z}}_{t,\nu}) \in \mathbb{D}^n \subset \mathbb{R}^{n \times d}$

temporally predictive image embeddings



The training process of the world model is the same as the original transformer encoder-decoder model [35]

[35] Vaswani et al., „Attention is All You Need“, arXiv:1706.03762, 2017

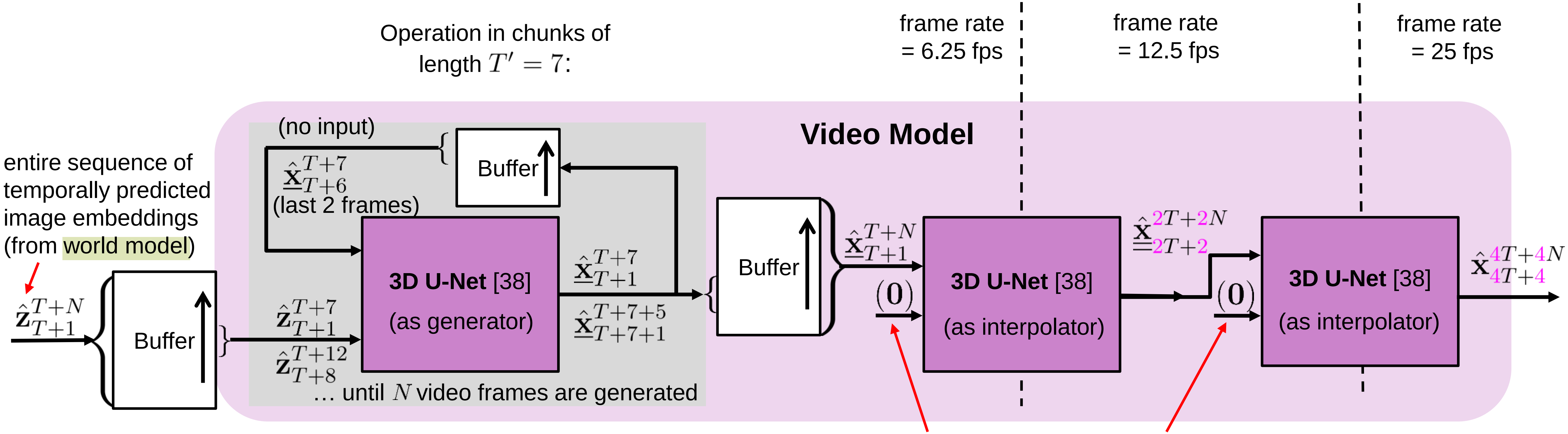
3. GAIA-1 Video Model (Inference)

The video generation model (3D U–Net) can be conditioned on image embeddings ($\hat{\mathbf{z}}$) and/or on images ($\underline{\hat{\mathbf{x}}}$, $\underline{\underline{\hat{\mathbf{x}}}}$).

When conditioned on image embeddings, it serves as a frame generator

When conditioned on images only, it serves as an interpolator

Operation in chunks of length $T' = 7$:



frame rate = 6.25 fps
frame rate = 12.5 fps
frame rate = 25 fps
Video Model
entire sequence of temporally predicted image embeddings (from world model)
$\hat{\mathbf{z}}_{T+1}^{T+N}$
Buffer
(no input)
$\underline{\hat{\mathbf{x}}}_{T+6}^{T+7}$
(last 2 frames)
Buffer
$\hat{\mathbf{z}}_{T+1}^{T+7}$
$\hat{\mathbf{z}}_{T+8}^{T+12}$
3D U-Net [38]
(as generator)
$\underline{\hat{\mathbf{x}}}_{T+1}^{T+7}$
$\underline{\hat{\mathbf{x}}}_{T+7+1}^{T+7+5}$
… until N video frames are generated
Buffer
$\underline{\hat{\mathbf{x}}}_{T+1}^{T+N}$
$(\mathbf{0})$
3D U-Net [38]
(as interpolator)
$\underline{\underline{\hat{\mathbf{x}}}}_{2T+2}^{2T+2N}$
$(\mathbf{0})$
3D U-Net [38]
(as interpolator)
$\hat{\mathbf{x}}_{4T+4}^{4T+4N}$

no embeddings $\hat{\mathbf{z}}$ are used as condition

first chunk: video generation

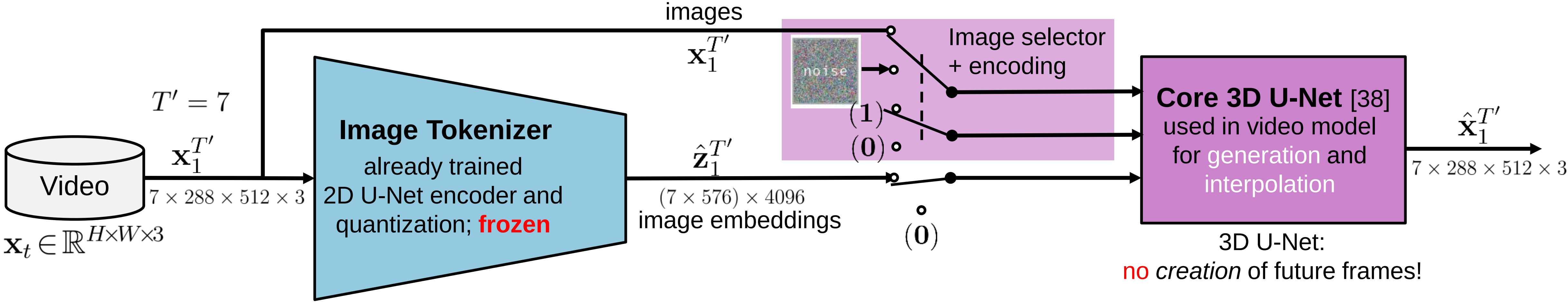
second chunk: AR video generation

[38] J. Ho, T. Salimans, A. Gritsenko, W. Chan, M. Norouzi, and D. J. Fleet, “Video Diffusion Models.” arXiv, Jun. 22, 2022.

3. GAIA-1 Video Model

Multitask training of 3D U-Net (in chunks of 7 frames)

In training, the video model is conditioned on a sequence of images and image embeddings encoded by the image tokenizer:

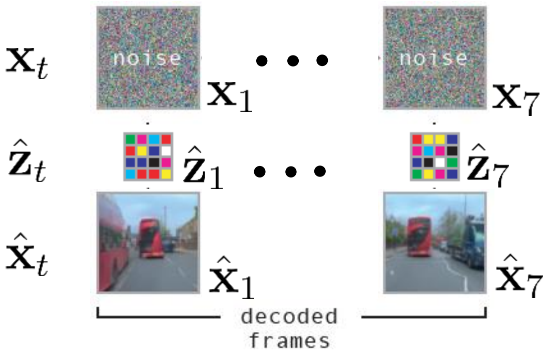


3D U-Net:
no *creation* of future frames!

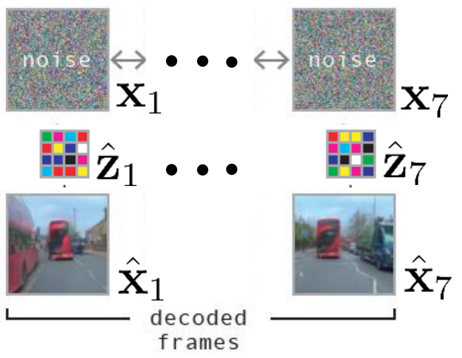
A single 3D U-Net model is trained on multiple tasks, conditioned on (masked) image embeddings and (masked) images
The selector positions are specified by the training task: (each task is equally represented in training)

(a) Image generation

(temp. layers deactivated)

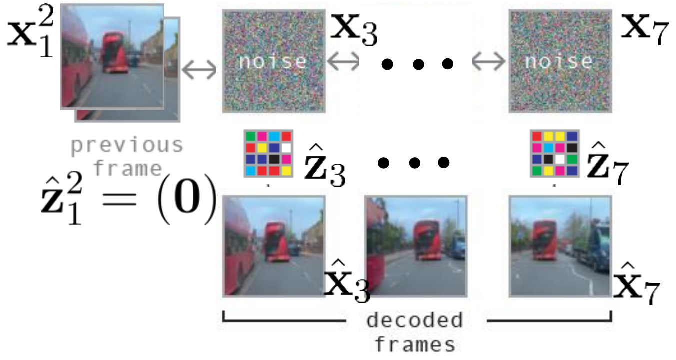


(b) Video generation



(c) Autoregressive video generation

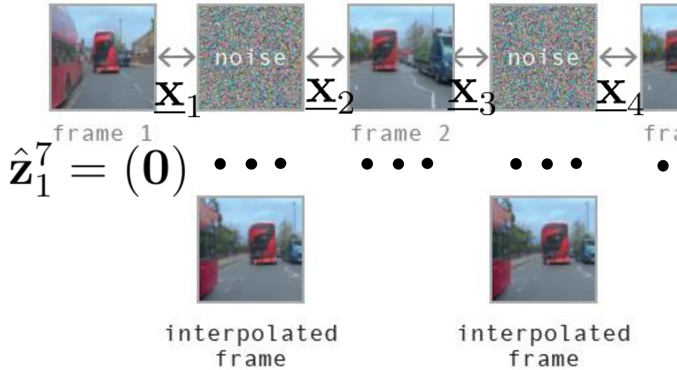
(2 old images and 5 new world model outputs)



frame rate = 6.25 fps

(d) Video interpolation

(every other image and all image tokens are masked)

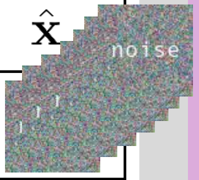


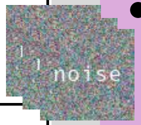
frame rate = 12.5 fps or 25 fps

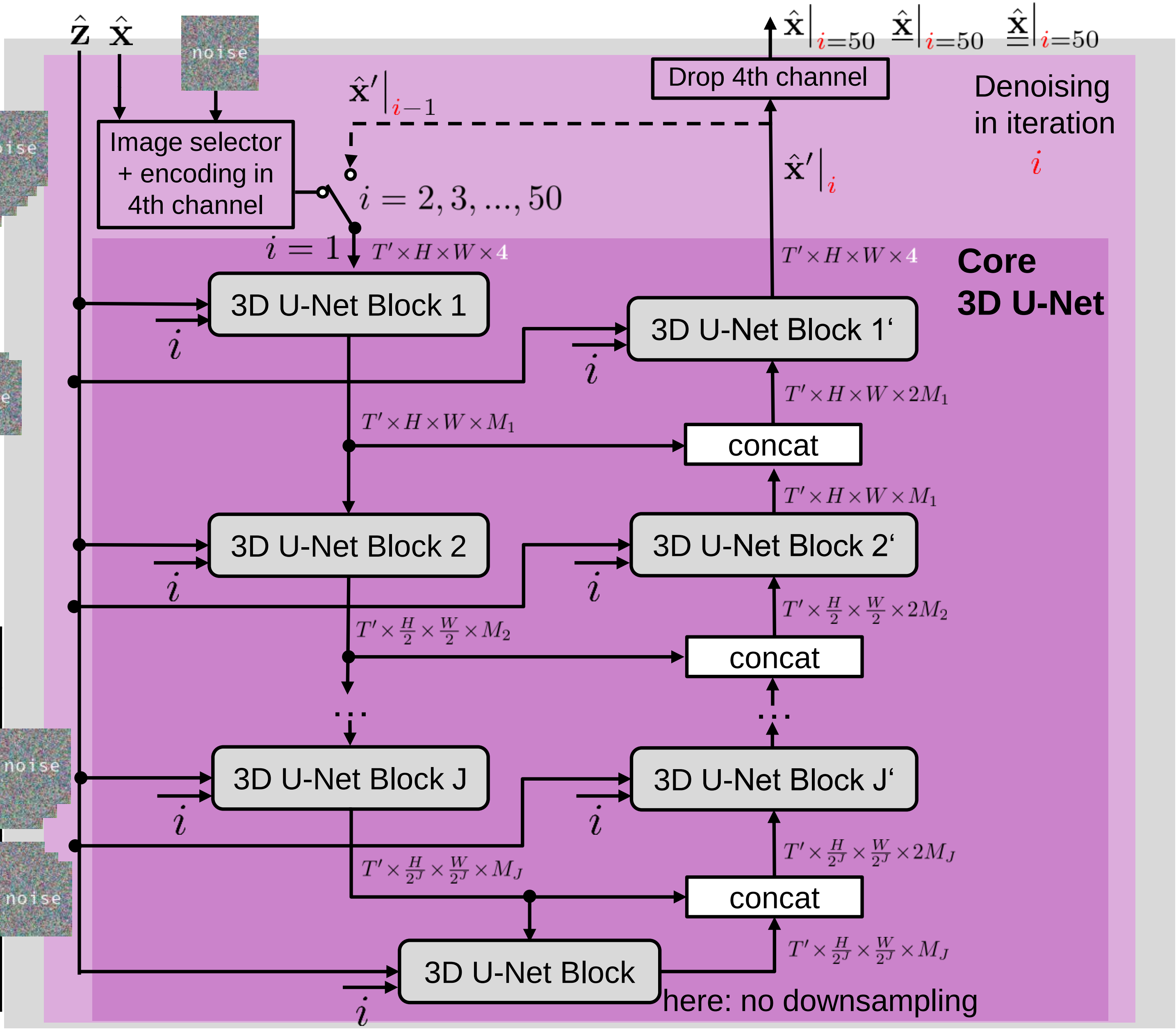
[38] J. Ho, T. Salimans, A. Gritsenko, W. Chan, M. Norouzi, and D. J. Fleet. Video diffusion models. In arXiv preprint, 2022.

3. GAIA-1 Video Model 3D U-Net [38]: Image Processing in Chunks of 7 Frames

Video generation process (by iterative denoising)

(AR) Video Generation (b) ((c))	Input data	
	$\hat{\mathbf{z}}$	$\hat{\mathbf{x}}$
1st chunk (b)	$\hat{\mathbf{z}}_{T+1}^{T+7}$	
2nd chunk (c)	$(\mathbf{0})$ $\hat{\mathbf{z}}_{T+8}^{T+12}$	$\underline{\hat{\mathbf{x}}}_{T+6}^{T+7}$
3rd chunk (c)	...	...

Video Interpolation (d)	Input data	
	$\hat{\mathbf{z}}$	$\hat{\mathbf{x}}$
1st chunk	$(\mathbf{0})$	$\underline{\hat{\mathbf{x}}}_{T+1}^{T+4}$
2nd chunk	$(\mathbf{0})$	$\underline{\hat{\mathbf{x}}}_{T+4}^{T+7}$ 
3rd chunk	...	...



$\hat{\mathbf{z}}$ $\hat{\mathbf{x}}$
noise
Image selector + encoding in 4th channel
$\hat{\mathbf{x}}'\big|_{i-1}$
$i = 2, 3, ..., 50$
$i = 1$
$T' \times H \times W \times 4$
Drop 4th channel
$\hat{\mathbf{x}}\big|_{i=50}$ $\hat{\mathbf{x}}\big|_{i=50}$ $\underline{\hat{\mathbf{x}}}\big|_{i=50}$
$\hat{\mathbf{x}}'\big|_{i}$
Denoising in iteration i
Core 3D U-Net
$T' \times H \times W \times 4$
3D U-Net Block 1
3D U-Net Block 1‘
i
$T' \times H \times W \times 2M_1$
$T' \times H \times W \times M_1$
concat
$T' \times H \times W \times M_1$
3D U-Net Block 2
3D U-Net Block 2‘
$T' \times \frac{H}{2} \times \frac{W}{2} \times 2M_2$
$T' \times \frac{H}{2} \times \frac{W}{2} \times M_2$
concat
3D U-Net Block J
3D U-Net Block J‘
$T' \times \frac{H}{2^J} \times \frac{W}{2^J} \times 2M_J$
$T' \times \frac{H}{2^J} \times \frac{W}{2^J} \times M_J$
concat
3D U-Net Block
$T' \times \frac{H}{2^J} \times \frac{W}{2^J} \times M_J$
here: no downsampling

The neural network architecture is a **3D U-Net with factorized spatial and temporal attention layers [38]**

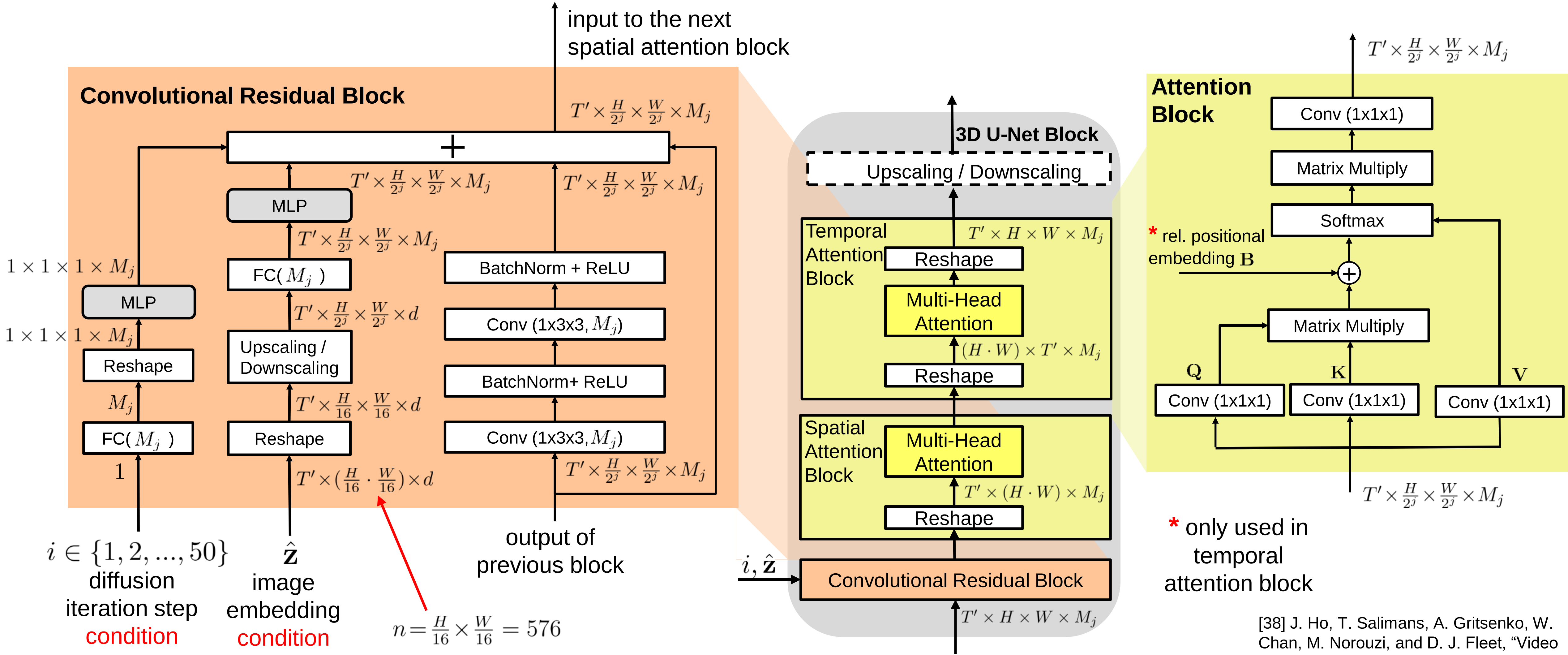
Image/video generation by iterative denoising in 50 steps $i = 1, 2, ..., 50$

M_j defines the feature dimensionality after a 3D U-Net block and can be configured

[38] J. Ho, T. Salimans, A. Gritsenko, W. Chan, M. Norouzi, and D. J. Fleet, “Video Diffusion Models.” arXiv, Jun. 22, 2022.

3. GAIA-1 Video Model 3D U-Net [38]

Building blocks


input to the next spatial attention block
Convolutional Residual Block
$T' \times \frac{H}{2^j} \times \frac{W}{2^j} \times M_j$
+
MLP
$T' \times \frac{H}{2^j} \times \frac{W}{2^j} \times M_j$
$T' \times \frac{H}{2^j} \times \frac{W}{2^j} \times M_j$
$T' \times \frac{H}{2^j} \times \frac{W}{2^j} \times M_j$
FC(M_j)
BatchNorm + ReLU
$1 \times 1 \times 1 \times M_j$
MLP
$T' \times \frac{H}{2^j} \times \frac{W}{2^j} \times d$
Conv (1x3x3, M_j)
$1 \times 1 \times 1 \times M_j$
Upscaling / Downscaling
Reshape
BatchNorm+ ReLU
M_j
$T' \times \frac{H}{16} \times \frac{W}{16} \times d$
FC(M_j)
Reshape
Conv (1x3x3, M_j)
1
$T' \times (\frac{H}{16} \cdot \frac{W}{16}) \times d$
$T' \times \frac{H}{2^j} \times \frac{W}{2^j} \times M_j$
$i \in \{1, 2, ..., 50\}$
diffusion iteration step
condition
$\hat{\mathbf{z}}$
image embedding
condition
output of previous block
$n = \frac{H}{16} \times \frac{W}{16} = 576$
3D U-Net Block
Upscaling / Downscaling
Temporal Attention Block
$T' \times H \times W \times M_j$
Reshape
Multi-Head Attention
$(H \cdot W) \times T' \times M_j$
Reshape
Spatial Attention Block
Multi-Head Attention
$T' \times (H \cdot W) \times M_j$
Reshape
$i, \hat{\mathbf{z}}$
Convolutional Residual Block
$T' \times H \times W \times M_j$
$T' \times \frac{H}{2^j} \times \frac{W}{2^j} \times M_j$
Attention Block
Conv (1x1x1)
Matrix Multiply
Softmax
* rel. positional embedding $\mathbf{B}$
Matrix Multiply
Q
K
V
Conv (1x1x1)
Conv (1x1x1)
Conv (1x1x1)
$T' \times \frac{H}{2^j} \times \frac{W}{2^j} \times M_j$
* only used in temporal attention block

[38] J. Ho, T. Salimans, A. Gritsenko, W. Chan, M. Norouzi, and D. J. Fleet, “Video Diffusion Models.” arXiv, Jun. 22, 2022.

3. GAIA-1

Datasets and Resources

Driving data has been collected in London, UK (2019-2023)

Training data: 4.700 hours at 25Hz (~420 million images)

Data balancing to account for geography and visually distinct weather conditions:

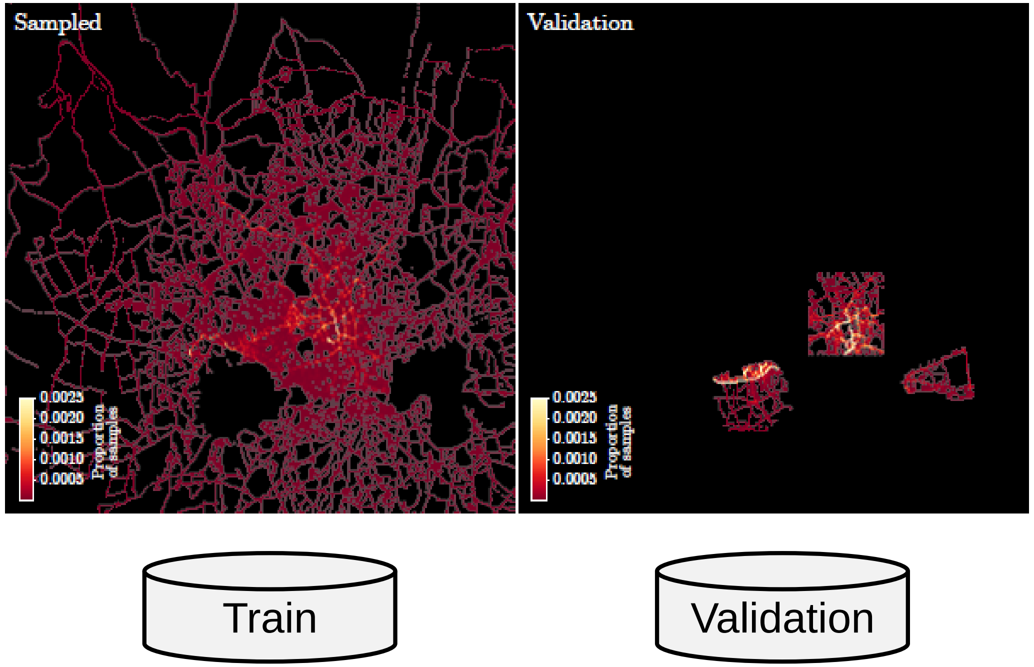
- Tokenizer balanced over latitude, longitude, weather conditions
- World model & video model balanced over latitude, longitude, weather, steering behaviour, speed behaviour

Validation data: 400 hours

Validation within strict predetermined geofences:

- 2 geofences with roads never seen during training
- 1 geofence around the main data collection routes but with runs not used during training

Driving data road map (London)



3. GAIA-1

Dataset Sizes

How does GAIA-1 training data compare to typical open access datasets in automotive research?

Name	# images	# annotated images	total length of videos [h]	# frames per second
CamVid	701	701	<1	1.00 - 15.00
KITTI	19,103	19,103	<1	10.00
Cityscapes	150,000	5,000	5	16.67
Waymo Open Perception	230,000	230,000	7	10.00
A2D2	392,556	41,277	<1	30.00
Caltech Pedestrian	1,000,000	250,000	10	30.00
nuScenes	1,200,000	40,000	15	11.67
SODA10M	10,000,000	20,000	27,833	0.10
BDD100K	120,000,000	100,000	1,111	30.00
GAIA-1 train+val	**420,000,000**	**?**	**4,700 + 400**	**25.00**

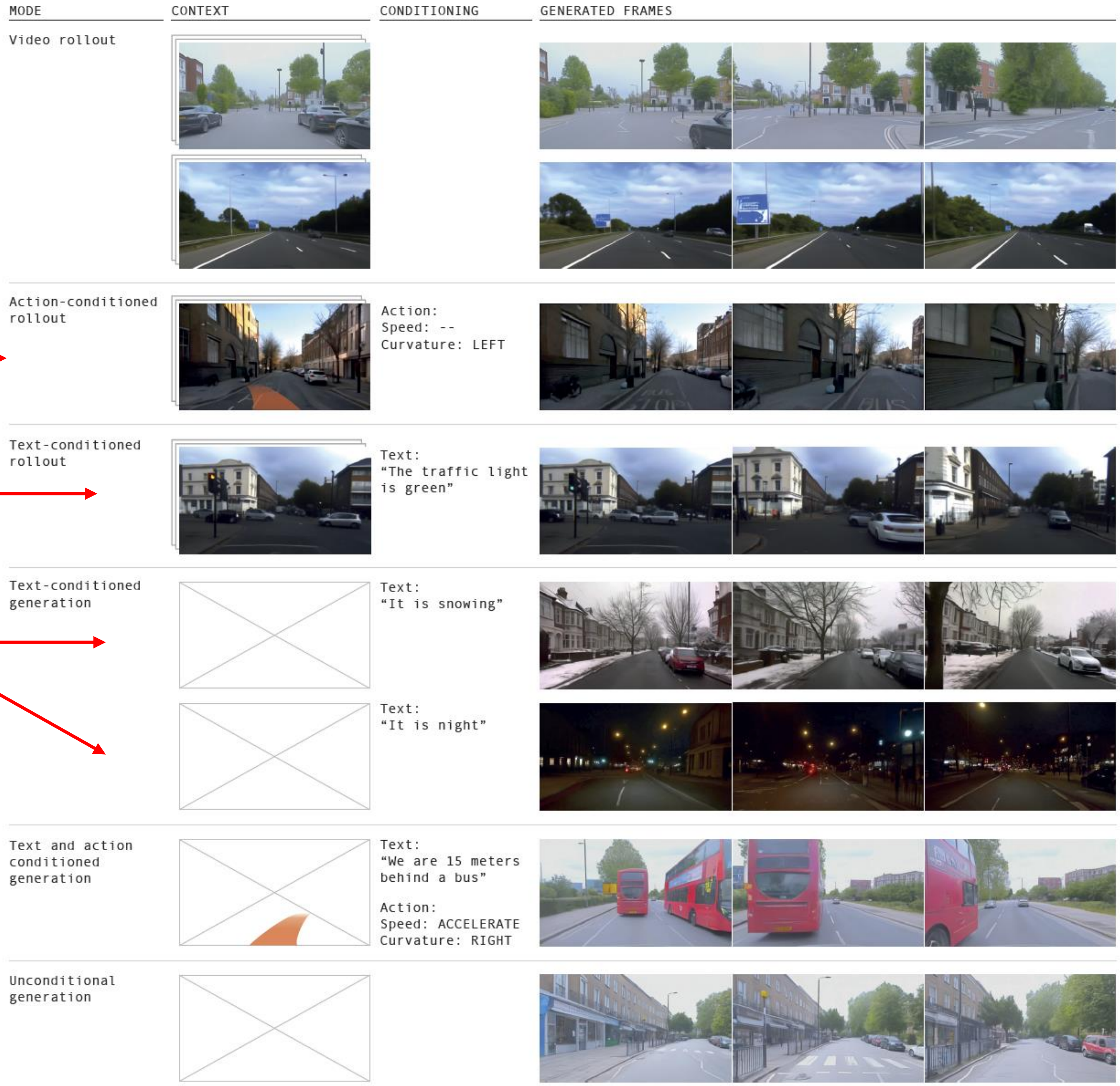
← BDD is useful!
Annotations for video attributes, include weather, scene, and time of day.

3. GAIA-1
Further Results …

Using text or (here:) actions, GAIA-1 can be forced to drive onto the pavement (never seen in training!)

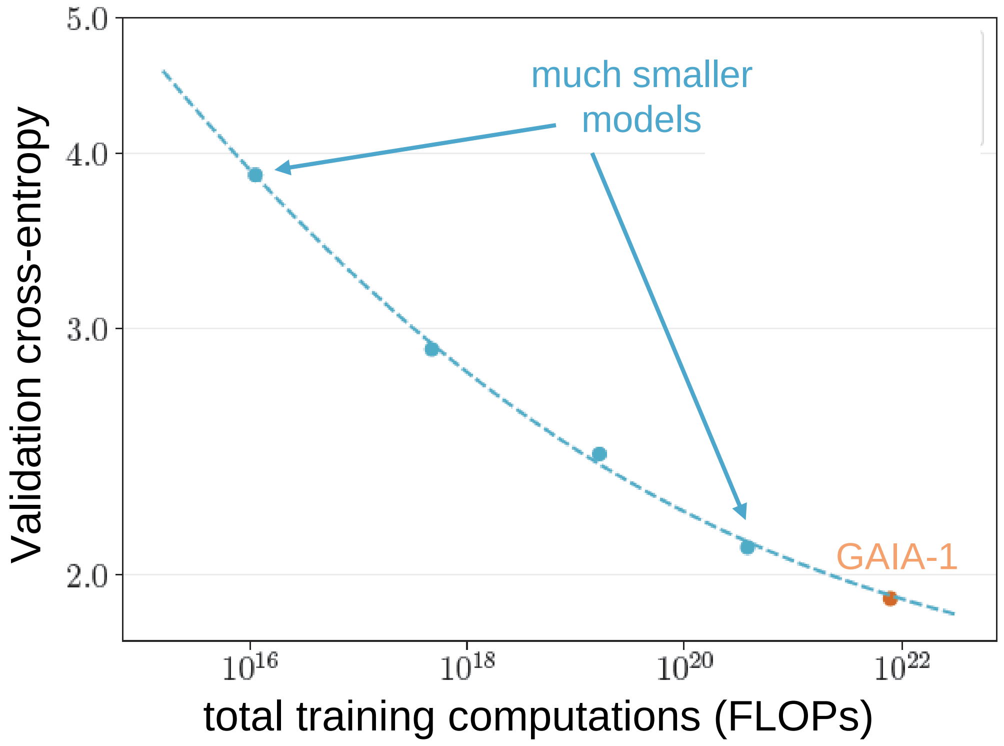
GAIA-1 knows: Green light means go ahead!

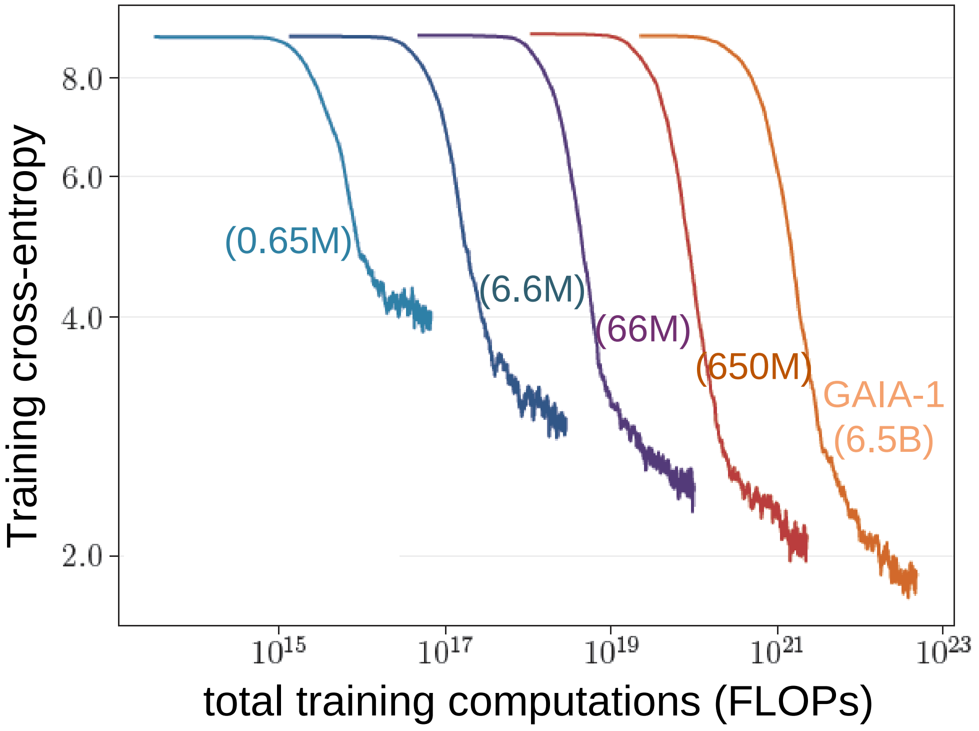
No video prompt, but text conditioning (here: weather, daytime)



3. GAIA-1

Scalability: World model evaluation with much smaller model size





GAIA-1 world model validation performance is predictable from smaller world models

GAIA-1 world model training performance gets better and better with a larger world model and the use of more data

[Hu, Anthony, et al. "GAIA-1: A Generative World Model for Autonomous Driving." *arXiv preprint arXiv:2309.17080* (2023)]

Conclusions

ASR: End2end automatic speech recognition achieves SOTA performance with attention-based encoder-decoder (AED) models

LLMs: (Large) language models (e.g., ChatGPT) achieve SOTA with attention-based decoder models

GAIA-1 achieves impressive results with an attention-based encoder-decoder (AED) world model

What we can learn:

Use standard separately trained tokenizers for each input modality; discretize patches of input images

Build multimodal foundational world models, integrating language and vision

Let the world model do the temporal prediction, and …

… let the video model reconstruct the output video in chunks

„Attention is all you need": It seems to be somewhat true…

[Vaswani et al., „*Attention is All You Need*", arXiv:1706.03762, 2017]

Thank you
for your attention …

Prof. Dr.-Ing. Tim Fingscheidt

t.fingscheidt@tu-bs.de

Slides on ResearchGate