FusionBrain Challenge 2.0

AI Journey Contest 2022

AGENDA

01 FusionBrain Approach

02 FusionBrain Challenge 2.0 (FBC2)
Motivation, task description

03 FBC2 Baseline. RUDOLPH

04 Technical Details
Public and Private test format, docker, metrics

05 eco2AI
Special category "The Most Sustainable Solution"

06 Questions

(Very!) Useful links

DS WORKS

FusionBrain Challenge 2.0

ai-forever/fbc2-aij2022
ai-forever/ru-dolph

sberbank-ai/RUDOLPH-2.7B-FBC2
sberbank-ai/RUDOLPH-2.7B
sberbank-ai/RUDOLPH-1.3B
sberbank-ai/RUDOLPH-350M

01

FusionBrain Approach

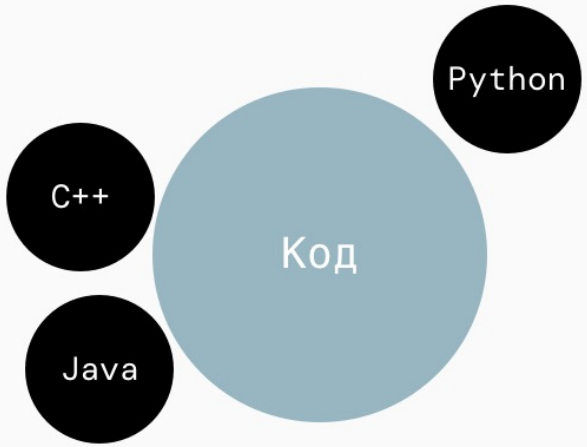
Мотивация: первое наблюдение

01 Человек – мультимодален
Нейронные сети (в большинстве случаев) – не

Что такое МОДАЛЬНОСТЬ?

Нет общепринятого определения

Можно попытаться формализовать это понятие, представив иерархическую структуру следующего рода



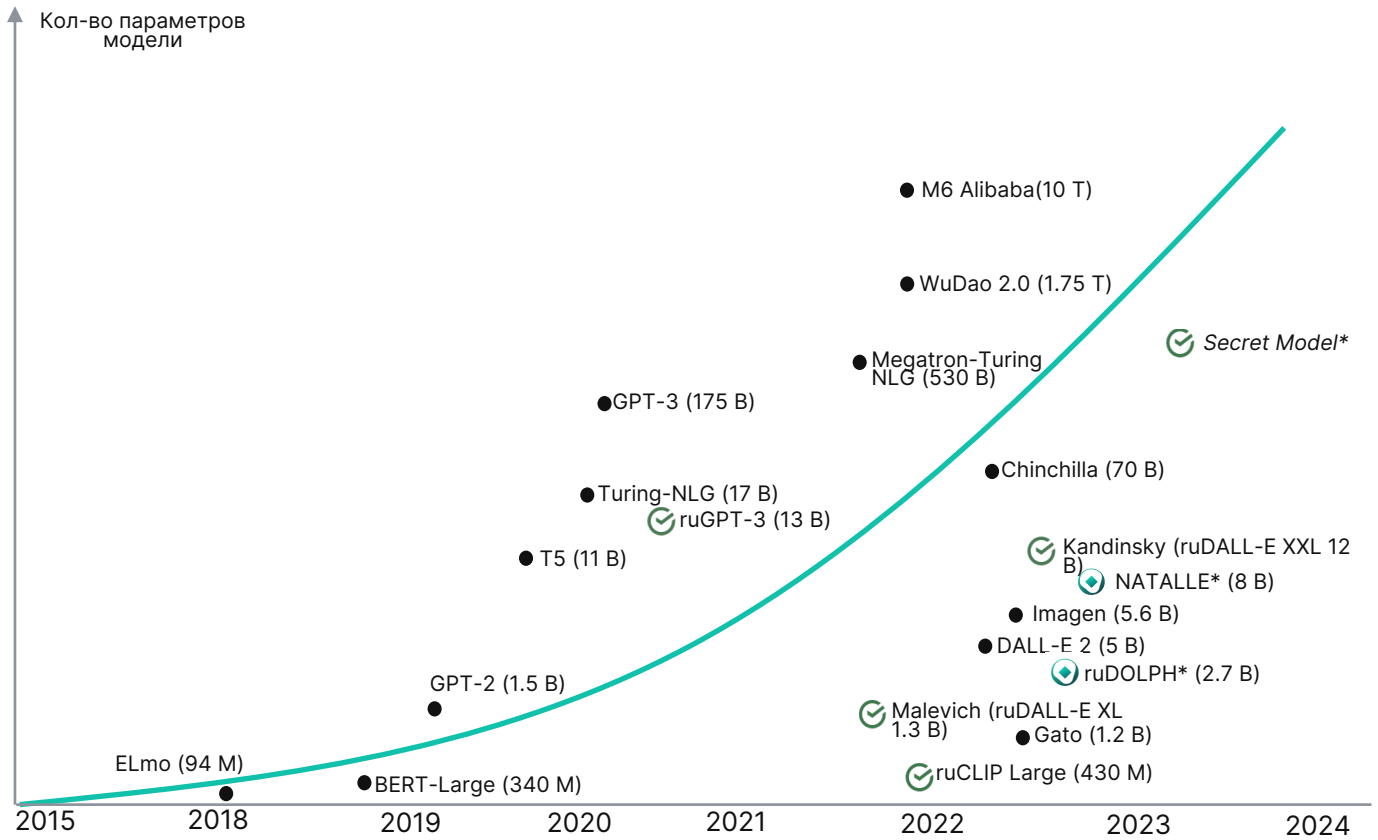
Мотивация: второе наблюдение

01 Человек – мультимодален
Нейронные сети (в большинстве случаев) – нет

02 Разработка новых мультимодальных и мультизадачных архитектур – современный **тренд** и большой научный и инженерный **вызов**. Это один из способов создания **AGI**

В **1.5 раза** больше публикаций **#multimodal #multitask** на arxiv за 2022, чем за аналогичный период 2021

Тренды в AI: мультимодальность, мультизадачность



* Планируется релиз

Разработка новых мультимодальных и мультизадачных архитектур – современный тренд и большой научный и инженерный вызов

Stanford: *Foundation Models*

OpenAI: GPT-3, CLIP, DALL-E, DALL-E 2

Google Brain: Imagen

Google Research: Parti

DeepMind: Gato, Flamingo

Обе картинки сгенерировал ИИ (Imagen)

Альпака стоит на дне бассейна, наполненного водой, перед объективом камеры, и смотрит в него, все происходит в солнечный день, реалистичное фото

Пейзаж гор на закате, горы теряются в дымке облаков, на переднем плане слева цветет сакура, реалистичное фото

Обе картинки сгенерировал ИИ (DALL-E 2)

Пикачу, выложенный плиткой

Авокадо в солнцезащитных очках танцуют и поют на гавайском луау, 3D-рендер

Опыт команд Сбера:
ruGPT-3, ruT5, mGPT, mT5, ruCLIP, ruDALL-E[1]

Модели в 1 модальности

текст

Модели в 2 модальностях

текст + изображения

Модели в 3+ модальностях

текст + изображения + видео + аудио + код + графы + временные ряды

[1] Источник: https://sbercloud.ru/ru/datahub/rugpt3family

Разрабатываем и обучаем фундаментальные модели

2 ноября

Malevich[2] (ruDALL-E XL)

1.3B

Генерация изображений по тексту на русском языке

Demo ruDALLE

Демо text-to-image модели, 1.3 млрд параметров

2021

19 января

ruCLIP[1]

430M

Zero-shot классификация картинок и текстов

ruCLIP Large [vit-large-patch14-336] exclusive

Модель-ранжировщик текстов и изображений, 430 млн параметров

2022

14 июня

Kandinsky[2] (ruDALL-E XXL)

12B

Генерация изображений по тексту на русском языке

ruDALL-E XXL Kandinsky

Большая модель (12 млрд. параметров) генерации изображений по текстовому описанию на русском языке.

Чебурашка в космосе

NATALLE 8.0B

8.0B

Диффузионная модель генерации изображений по тексту на 100 языках (мультиязычная)

RUDOLPH 2.7B

2.7B

Модель, которая решает 25+ задач в модальностях изображения и тексты на русском языке

Timeseries Transformer

1.3B

Фундаментальная модель для работы с транзакционными данными, кликстримами итд

Secret Model

X B

Большой мультимодальный трансформер, который решает Y+ задач в N модальностях

2023

[1] Shonenkov, Alex, et al. “RuCLIP — New Models and Experiments”, 2022.
[2] Dimitrov, Denis, et al. “RuDALL-E — New Zero-Shot Text-to-Image Generation”, 2022.

Мотивация: третье наблюдение

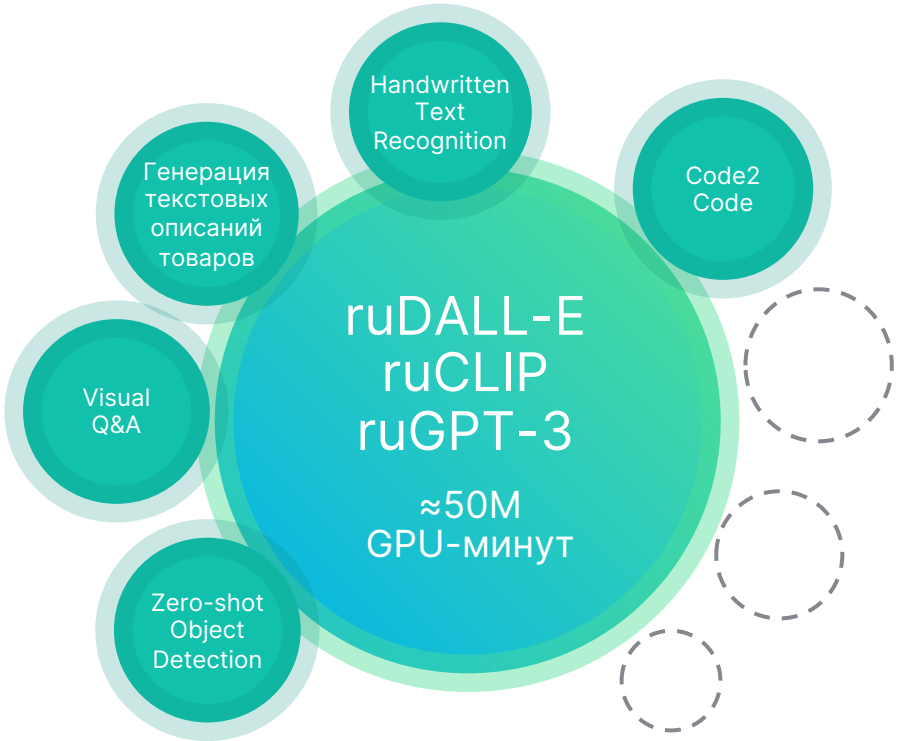
01 Человек – мультимодален
Нейронные сети (в большинстве случаев) – нет

02 Разработка новых мультимодальных и мультизадачных архитектур – современный **тренд** и большой научный и инженерный **вызов**. Это один из способов создания **AGI**

03 Благодаря использованию предобученных фундаментальных моделей (foundation models) можно добиться существенной **экономии** ресурсов (GPU-минут) при обучении прикладных моделей

Интуиция: экономия ресурсов (GPU-минут) при обучении прикладных моделей

VS



≈1300M
GPU-минут

Решение каждой задачи с нуля:
много мощностей и данных
на решение каждой

≈100M
GPU-минут

Использование предобученных моделей: небольшое количество вычислительных ресурсов и данных для дообучения

FusionBrain Challenge (1.0) (01.10.2021 – 06.11.2021)

Tasks

An unified architecture that solves all tasks together (at least 25% of the total parameters)

01

Code2code Translation

The task of translating from the Java programming language (static typing) to the Python programming language (dynamic typing)

Private test: **322**

02

Handwritten Text Recognition

The task of handwriting recognition on an image (Russian + English)

Private test: **12 556** (5 494 in English, 7 062 in Russian)

03

Zero-shot Object Detection

The task of detecting objects in an image upon request, which is a description in natural language (Russian + English)

Private test: **827** pictures

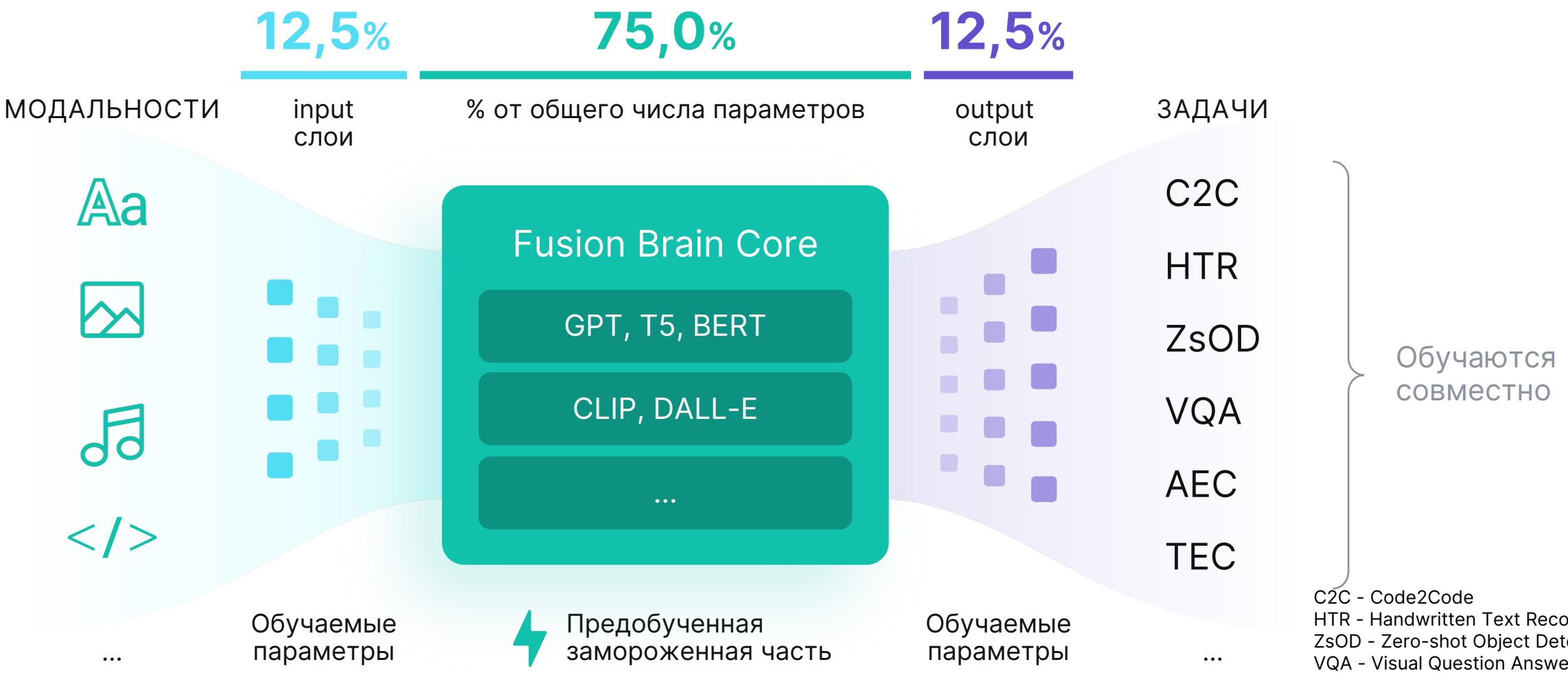
04

Visual Question Answering

The task of generating an answer to a question from a picture (Russian + English)

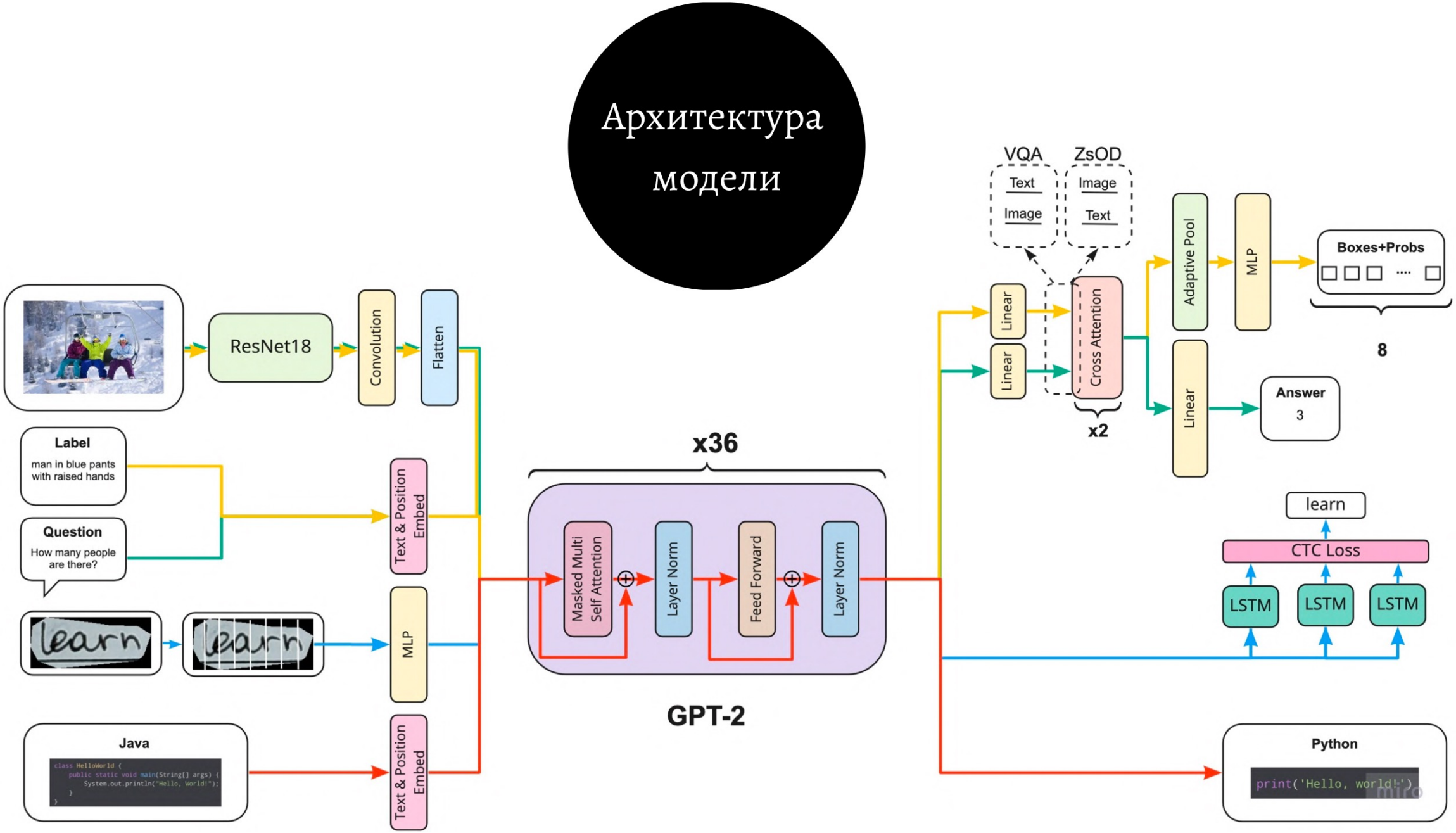
Private test: 1 000 pictures, 6 000 questions

FusionBrain: концепт



C2C - Code2Code
HTR - Handwritten Text Recognition
ZsOD - Zero-shot Object Detection
VQA - Visual Question Answering
AEC - Audio Emotion Classification
TEC - Text Emotion Classification

Архитектура модели



Result

Many Heads but One Brain: an Overview of Fusion Brain Challenge on AI Journey 2021

Daria Bakshandaeva[1,*], Denis Dimitrov[1,3,*], Alex Shonenkov[1], Mark Potanin[1], Vladimir Arkhipkin[1], Denis Karachev[1], Vera Davydova[1], Anton Voronov[2], Mikhail Martynov[1], Natalia Semenova[1], Mikhail Stepnov[1], Elena Tutubalina[1], Andrey Chertok[1,2], Aleksandr Petiushko[2,3]
[1] Sber AI, [2] Artificial Intelligence Research Institute, [3] Lomonosov Moscow State University
Moscow, Russia
{DDBakshandaeva,Dimitrov.D.V,AVShonenkov,potanin.m.st}@sberbank.ru,arkhipkin.v98@gmail.com,denis.karachev@ocrv.ru,
VFeDavydova@sberbank.ru,Voronov@airi.net,{mmmartynov,NAlSemenova,mistepnov,EVTutubalina}@sberbank.ru,
{Chertok,Petiushko}@airi.net

training setup	C2C CodeBLEU	HTR Acc	ZsOD F1	VQA Acc	Overall
Single-task	0.34	**0.63**	0.17	0.25	1.39
Fusion	**0.39**	0.61	**0.21**	**0.30**	**1.51**

TABLE I
PRIVATE SCORES FOR DIFFERENT TRAINING STRATEGIES

training setup	Training time (hours)	Training params	CO2 (kg)
Single-task	215.0	3,283,978,882	39.34
Fusion	**150.5**	**988,272,474**	**27.45**

TABLE II
TOTAL PARAMETERS SUMMARIZED FOR ALL 4 TASKS

Alexandre Lacoste, Alexandra Luccioni, Victor Schmidt, and Thomas Dandres. Quantifying the carbon emissions of machine learning. *arXiv preprint arXiv:1910.09700*, 2019.

Private leaderboard

Public leaderboard | Private leaderboard

Rank	Team name	Submissions	Score	Medals
1	qbic	1	1.680	
2	orzhan	1	1.032	
3	SpaceDoge (unitask)	1	0.910	
4	Arasaka	1	0.907	
5	Magic City	1	0.817	
6	dwayne Scala JSON	1	0.766	
7	mihtw	1	0.614	
8	alxmamaev	1	0.613	
9	DeepPavlov (out-of-competition)	1	0.612	
10	sberaiooc	1	0.548	

13 countries
400+ participants
1000+ submissions

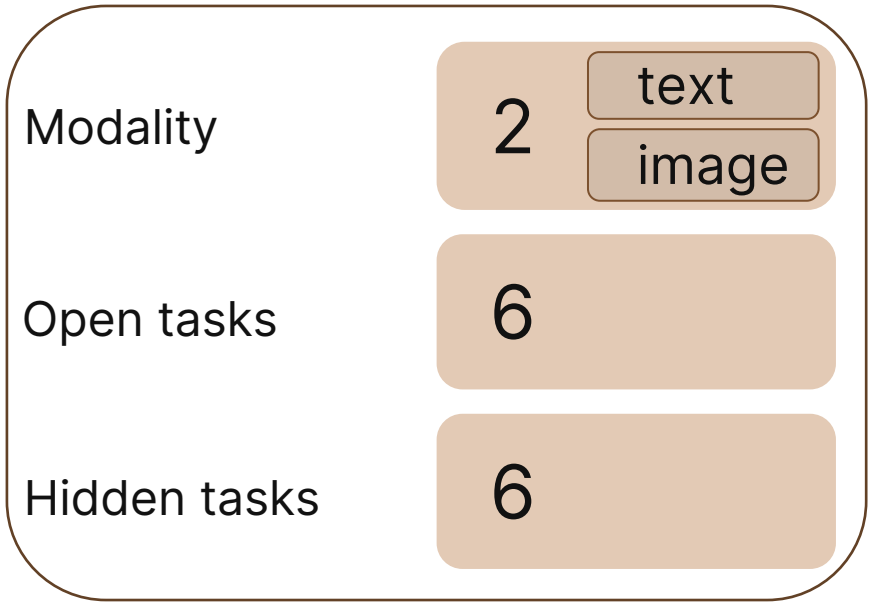
02

FusionBrain Challenge 2.0

FusionBrain Challenge 2.0

AI Journey 2022

- Bi-modality (**image** and **text**)
  - A model should be able to operate within two modalities simultaneously.
- Multi-tasking (half of the tasks is **open** and the other half is **hidden**)
  - A model should generalize for hidden tasks via open-tasks training.
  - A model should be able to solve various tasks by following natural language instructions.
- All the tasks are given in natural language (Russian).



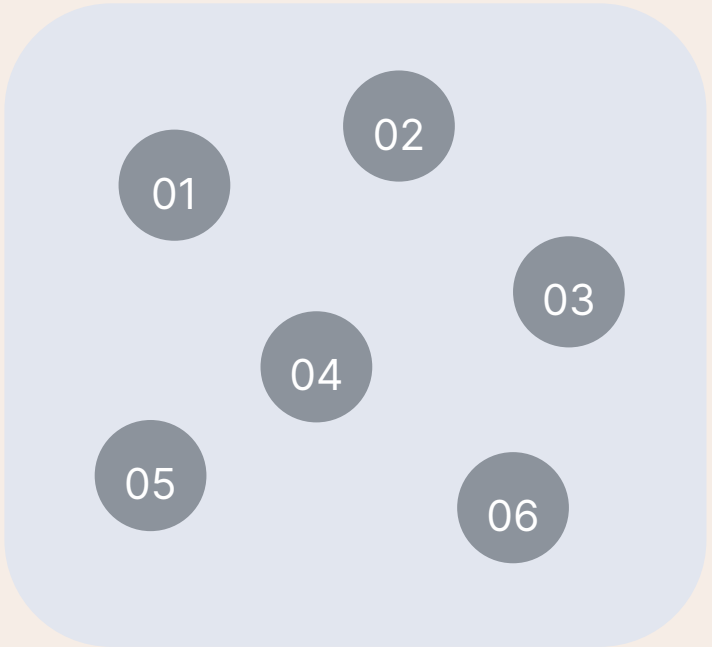
FusionBrain Challenge 2.0

AI Journey 2022

Open tasks

1. Text QA
2. Mathematical QA
3. Image Generation
4. Image Captioning
5. Visual QA
6. Text Recognition in the Wild

Hidden tasks



All the tasks are defined in natural language.
Some tasks are in the hidden form – there is no information about them.

Open Tasks Description

Text QA

Text QA is a NLP task that checks a model's ability to read and understand a piece of text in Russian by answering questions about it.

Type of task: **text-to-text**

Input format:

Input: 1. Task instruction 2. Passage 3. Question 4. Options (optionally)	**Input:** 1. Task instruction + Passage 2. Question 3. Options (optionally)	**Input:** 1. Passage 2. Question 3. Options (optionally)
Output: 1. Answer (text)		

Text QA Sample

Input: task instruction; passage; question

Task instruction: Определи правильный ответ по тексту.

Passage: Улица Авиаконструктора Сухого (название с 2004 года) — улица в Северном административном округе города Москвы на территории Хорошёвского района. Пролегает от Проектируемого проезда № 6161 до пересечения с юго-восточной частью взлётно-посадочной полосы Ходынского Поля. Название утверждено 30 марта 2004 года в честь знаменитого советского авиаконструктора Павла Осиповича Сухого (1895—1975).

Question: Какая улица пролегает от Проектируемого проезда № 6161 до пересечения с юго-восточной частью взлётно-посадочной полосы Ходынского Поля?

Answer: Авиаконструктора Сухого

Text QA. Training Data

Sber Question Answering Dataset (SberQuAD)


```
{
    "context": "Первые упоминания о строении человеческого тела встречаются в Древнем Египте...",
    "id": 14754,
    "qas": [
        {
            "id": 60544,
            "question": "Где встречаются первые упоминания о строении человеческого тела?",
            "answers": [{"answer_start": 60, "text": "в Древнем Египте"}],
        }
    ]
}
```


Prepared dataset (*textqa* folder)

train	validation
45 328	3 000

Text QA. Training Data

additional

Russian Multi-Sentence Reading Comprehension (MuSeRC)


```
{
        "id": 397,
        “text”: “...  (13) Они опередили своих основных соперниц - немок - всего на 0,3 секунды.”
        "questions": [
                {
                    "question": "На сколько секунд женская команда опередила своих соперниц?",
                    "answers": [
                       { "text": "Всего на 0,3 секунды.", "label": 1 },
                       { "text": "На 0,3 секунды.", "label": 1 },
                       { "text": "На секунду.", "label": 0 }, ...
                    ],
                    "idx": 0
                    }]
}
```


~6 000 samples

Binary classification task

Mathematical QA

Mathematical QA is a task to check the model's ability to solve mathematical equations.

Range of math tasks: first-order linear equations and their systems, comparison tasks.

Type of task: **text-to-text**

Input format

Input: 1. Task instruction 2. Math equation 3. Question 4. Options	**Input:** 1. Task instruction + Math equation 2. Question 3. Options	**Input:** 1. Math equation 2. Question 3. Options
Output: 1. Answer (text or correct option marker)		

Mathematical QA Sample

Input: task instruction + math equation; question; options

TI + ME: Решите уравнение. Если 6 умножить на y и вычесть 54, то получится -72.
Question: Чему равно y?
Options: a) -4; b) -20; c) -5; d) -3

Answer: d

Mathematical QA. Training Data

Mathematics Dataset
DeepMind (*Saxton et al., 2019*)
Dataset for Evaluation of Mathematical Reasoning Abilities in Russian (*Nefedov, 2020*)

Prepared dataset (*mathqa* folder)

Dataset preparation steps:

- algebra → 'linear_1d', 'linear_2d' modules
- Data augmentation ('+', '-', '*', '/', '=' → 'плюс'/'прибавить', 'минус', 'умножить', 'делить'

train	validation
100 000	3 000

Image Generation

Image Generation is a task for generating images based on textual description.

Type of task: **text-to-image**

Input format:

Input: 1. Task instruction 2. Image caption	**Input:** 1. Task instruction + Image caption
Output: 1. Generated image (path to the image)	

Image Generation Sample

Input: task instruction; image caption

Task instruction: Сгенерируй изображение по тексту.

Passage: На столе из светлого дерева стоит тарелка с яичницей из трех яиц, рядом белая чашка кофе.

Answer: images/00000.jpg

Image Generation. Training Data

Training data: “text-image” pairs; images are retrieved from the COCO dataset with captions translated in Russian with automatic translator.

Prepared dataset (*generation* folder)

train	validation
82 783	3 000

Image Captioning

Image Captioning is the task of describing the content of an image in words.

Type of task: **image-to-text**

Input format

Input:
1. Task instruction
2. Path to image

Output:
1. Image caption (text)

Image Captioning Sample

Input: task instruction; path to image

Task instruction: Что изображено на картинке?

Image: *images/00001.jpg*

Answer: Детский светлый торт на день рождения.

Image Captioning. Training Data

Training data: “image-text” pairs; images are retrieved from the COCO dataset with captions translated in Russian by automatic translator.

Prepared dataset (*captioning* folder)

train	validation
82 783	3 000

Visual QA

Visual QA is a task that aims to answer questions based on an image.

Type of task: **image-to-text**

Input format

Input: 1. Task instruction 2. Question 3. Path to image 4. Options (optionally)	**Input:** 1. Task instruction + Question 2. Path to image 3. Options (optionally)	**Input:** 1. Question 2. Path to image 3. Options (optionally)
Output: 1. Answer (text or correct option marker)		

Visual QA Sample

Input: question; path to image

Question: Какого цвета плакат на ограждении?

Image: images/00002.jpg

Answer: желтого

Visual QA. Training Data

Training dataset collected from Visual Genome dataset with open-ended questions about images and the corresponding ground truth answers. The questions and the answers are translated into Russian by automatic translator.

Prepared dataset (*vqa* folder)

train	validation
85 759	3 000

Text Recognition in the Wild

Text Recognition in the Wild (TRitW) is a task to recognize text within images captured in the wild (traffic signs, billboards, etc.)

Type of task: **image-to-text**

Input:
1. Task instruction
2. Image (*path*)

Output:
1. Recognized text

Text Recognition in the Wild Sample

Input: task instruction; path to image

Question: Распознай текст на изображении.

Image: images/00003.jpg

Answer: супермаркет недвижимости

Text Recognition in the Wild. Training Data

Training data: “image-text” pairs; images are retrieved from the COCO dataset with captions translated in Russian by automatic translator.

Prepared dataset (*text_recognition* folder)

START (**S**yn**T**hesized and **A**nnotated dataset for **T**ext **R**ecognition) dataset can be used for training. The dataset contains 140 000 images (retrieved from the COCO dataset) with overlayed synthetic text in Russian (situated in various parts of the images, with different colors and transparency, etc.) and 40 312 real urban photos with various text labels both in Russian (mainly) and English.

train	validation
40 312	3 000

Additional dataset
SberIDP Text in the Wild dataset contains 17 000 monochrome images with some text in Russian.

03

Baseline Solution RUDOLPH

RUDOLPH

RUDOLPH*: RUssian **D**ecoder **O**n **L**anguage **P**icture **H**yper-tasking

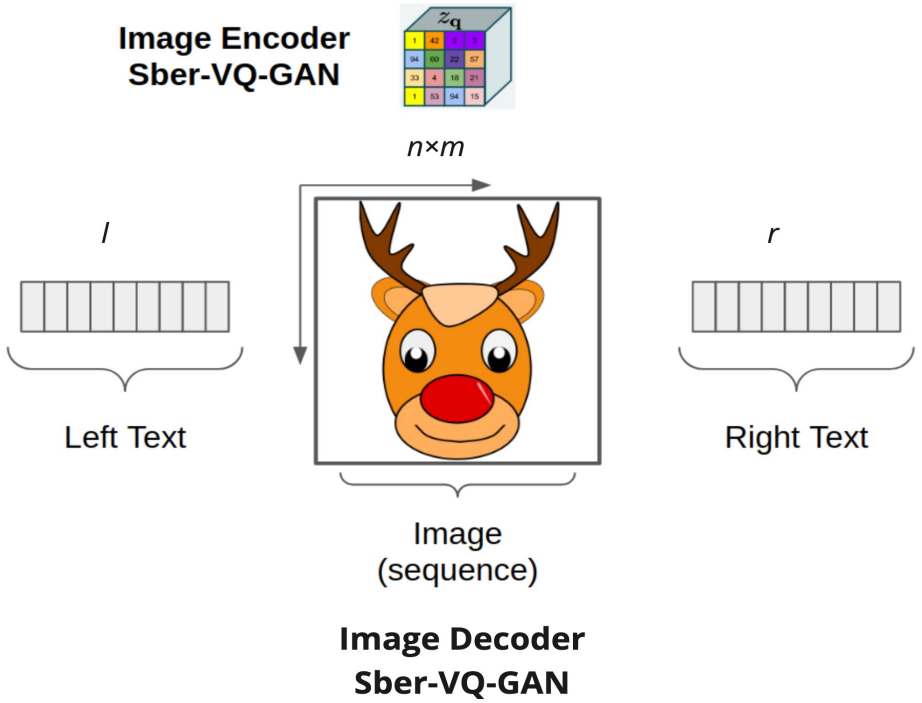
Hyper-tasking model for solving a range of tasks in two modalities (text, image)

text-to-image (t2i), image-to-text (i2t), text-to-text (t2t).

3 versions:

350M: n=64, m=64, k=16, l=16
1.3B: n=128, m=128, k=32, l=32
2.7B: n=384, m=128, k=24, l=24
2.7B-FBC2: n=384, m=128, k=24, l=24
(fine-tuned on 6 open tasks of FBC2)

Implements decoder block of the Transformer model.



*RUDOLPH github

RUDOLPH Architecture

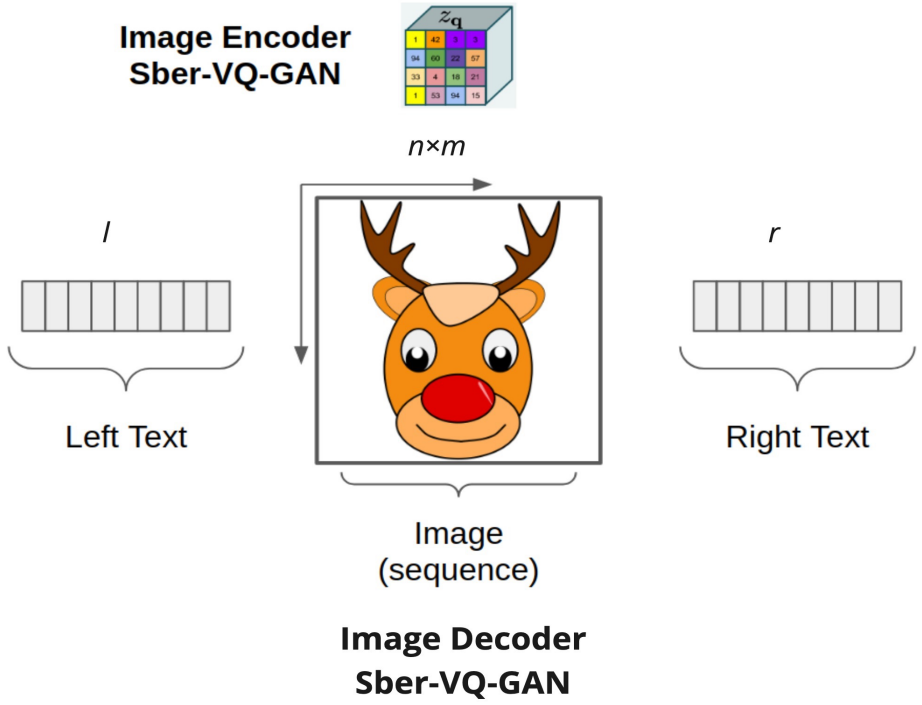
Implements decoder block of the Transformer model.

Model’s parameters (350M/1.3B/2.7B)

- Number of hidden layers: 24/24/32
- Dimensionality of the hidden layer: 1024/2048/2560
- Number of attention heads: 16/16/32

Text is tokenized and provided as an input sequence of tokens.

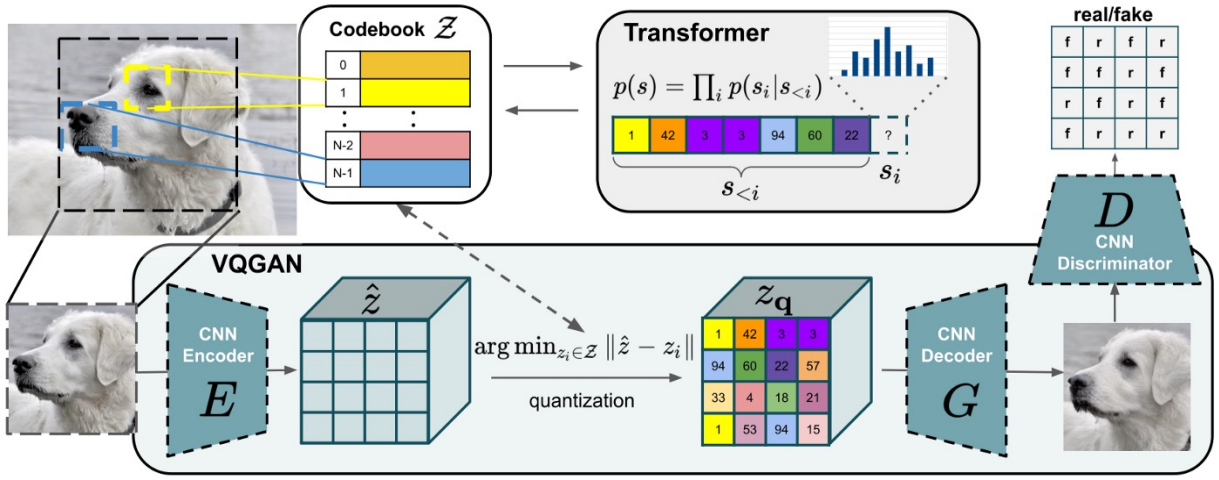
Image is processed by Sber-VQGAN encoder and provided to the model as flatten sequence of the image tokens.



Images Processing

An input image goes through Sber-VQGAN encoder.
To get an output image, image tokens are processed by Sber-VQGAN decoder.

VQGAN
Vector Quantized Generative Adversarial Networks



Sber-VQGAN

VQGAN with Gumbel Quantization fine-tuned on additional dataset from different domains.

Inception Score (IS ↑)

domain/model	VAE	16384	gumbelf8	SBER-gumbelf8	Original
all	11.133	13.647	15.203	**15.316**	15.278
indoor	9.769	10.744	11.707	11.688	11.638
kitchen	9.726	11.354	12.333	12.152	11.813
appliance	5.705	6.024	6.154	**6.199**	5.890
electronic	7.830	9.509	9.712	9.606	9.497
furniture	10.861	13.346	14.500	**14.531**	14.592
outdoor	8.163	9.520	10.668	10.293	10.451
sports	7.467	8.544	8.814	**8.841**	8.962
food	7.954	8.725	9.390	**9.434**	9.191
vehicle	10.527	12.947	14.240	**14.559**	14.233
animal	11.933	14.249	15.999	15.879	15.857
accessory	9.399	11.687	13.117	**13.388**	13.228
person	13.752	17.794	20.048	**20.420**	20.600
face	11.903	14.987	16.986	**17.489**	17.584
text	14.902	18.457	21.396	**21.292**	21.131

Sber-VQGAN


original
VAE
16384
gumbelf8
SBER-gumbelf8
original
VAE
16384
gumbelf8
SBER-gumbelf8
Комиссионный магазин
СКУПКА 96
REMPROFI
original
VAE
16384
gumbelf8
SBER-gumbelf8

RUDOLPH Pre-Training

Autoregressive training.

Predict next tokens based on the previous context (masked):

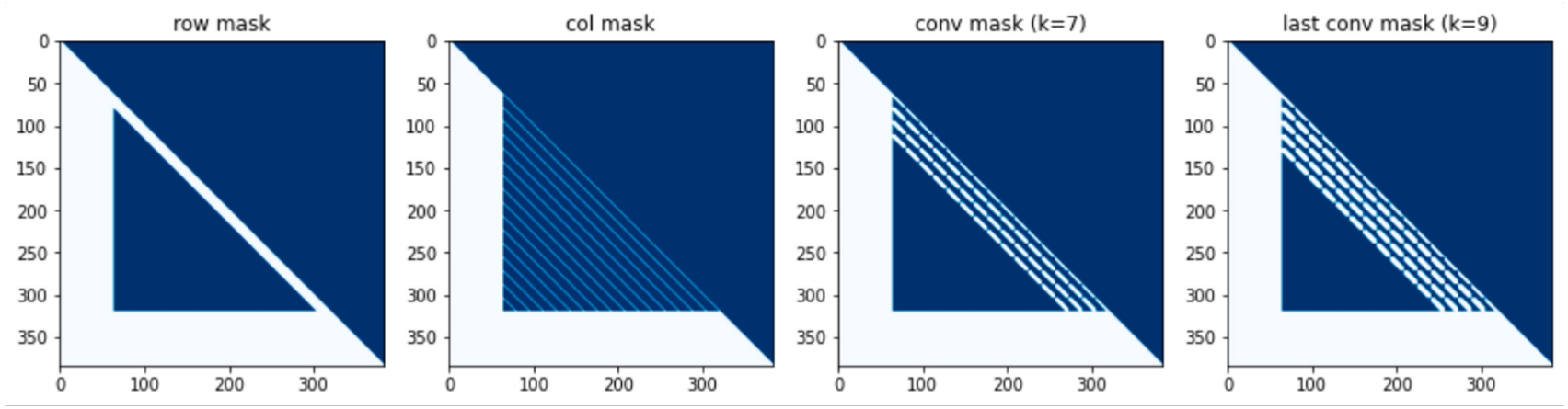
- **t2i**: image generation based on textual description
- **i2t**: caption generation based on the image
- **t2t**: language modeling (only in the left tokens)

Training data:

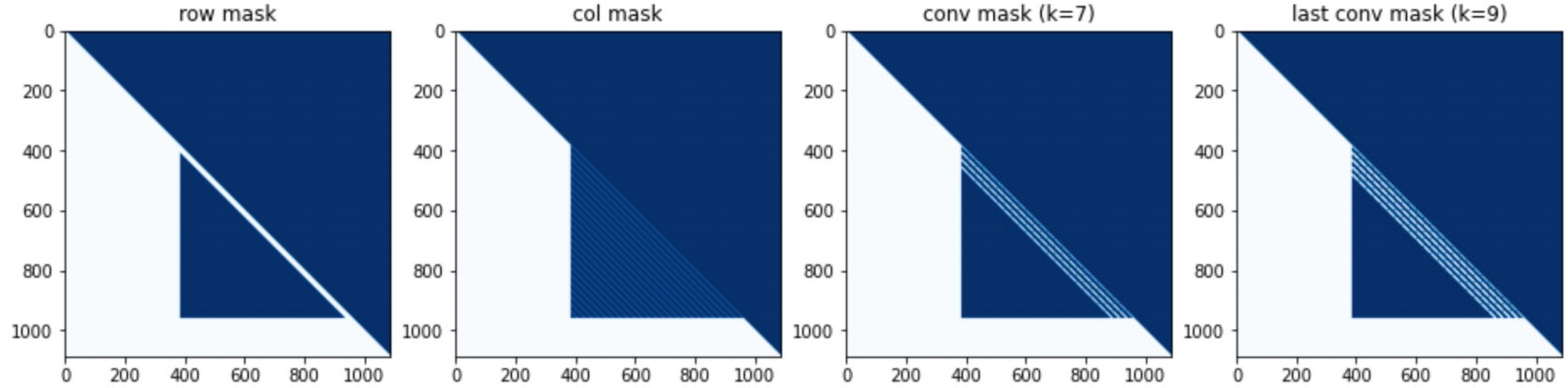
- **i2t/t2i**: 119M text-image pairs
- **t2t:** 60M text pieces

Attention Masks

350M:

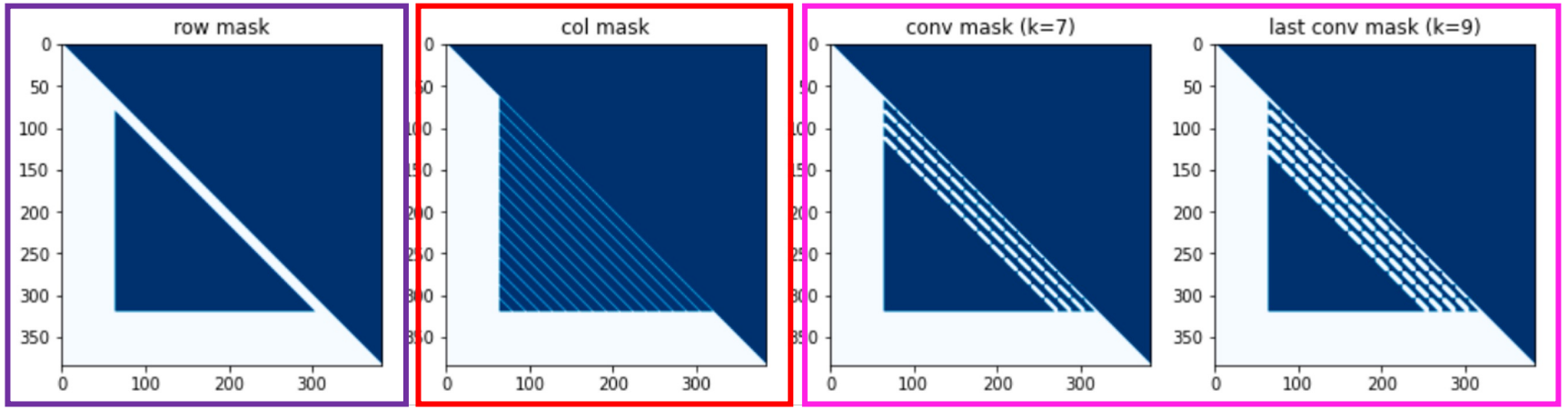


2.7B:

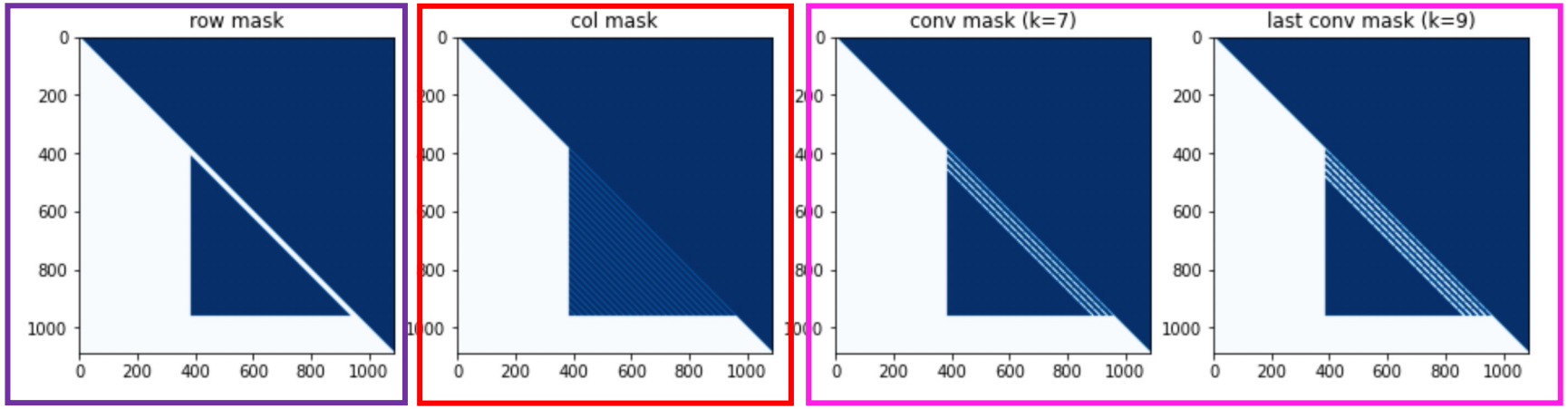


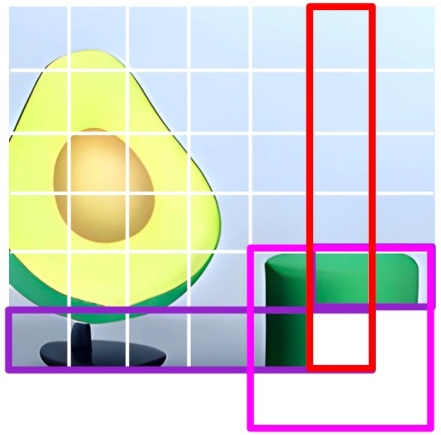
Attention Masks

350M:



2.7B:





Special Tokens for Task Understanding

Pre-training (75/25)

<LT_T2I> - *text-to-image*

<LT_I2T> - *image-to-text*

<LT_T2T> - *text-to-text*

<RT_I2T> - *image-to-text*

<LT_UNK>

<RT_UNK>

Pre-training (75/25)

<BOS> <LT_T2I> *left_text_tokens* | *image_tokens* | **<RT_I2T>** *right_text_tokens* **<EOS>**

<BOS> <LT_UNK> *left_text_tokens* | image_tokens | **<RT_I2T>** *right_text_tokens* **<EOS>**

<BOS> <LT_T2I> *left_text_tokens* | image_tokens | **<RT_UNK>** *right_text_tokens* **<EOS>**

Special Tokens for Tasks Understanding

Pre-training (75/25)

Task-specific tokens:

<LT_T2I> - *text-to-image*

<LT_I2T> - *image-to-text*

<LT_T2T> - *text-to-text*

<RT_I2T> - *image-to-text*

Tokens for the unknown tasks:

<LT_UNK>

<RT_UNK>

Fine-tuning (75/25)

Tokens for open tasks:

<LT_TQA>

<RT_TQA>

<LT_MQA>

<RT_MQA>

<LT_VQA>

<RT_VQA>

<LT_CAP>

<RT_CAP>

<LT_GEN>

<LT_REC>

<RT_REC>

Special Tokens for Tasks Understanding

During inference the model has no a-priori information about the task.

Add *unknown* tokens for each input sample:

- `<LT_UNK>`
- `<RT_UNK>`

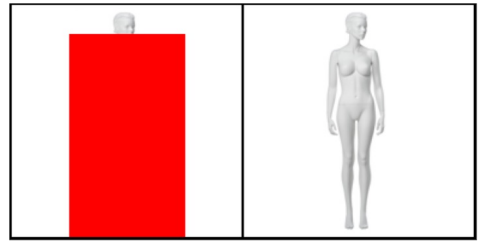
Zero-Shot Performance of RUDOLPH*

Image captioning

- на картинке собака породы бигль
- на картинке бигль с большими ушами и биглем с характером собаки
- на картинке бигль порода
- на картинке бигль голова
- на картинке собака бигль

Image inpainting

- *красивое платье синего цвета на манекене*
- *на манекене надеты синие джинсы и белая футболка*



**jupyter on zero-shot RUDOLPH performance is on github*

Fine-Tuned RUDOLPH for FBC2

- Text QA – SberQuAD – 45 328 “passage-text-question” triples
- Mathematic QA – DeepMind Mathematics Dataset – 100 000 samples
- Image Generation – 82 783 “text-image” pairs (COCO dataset with translated captions)
- Image Captioning – 82 783 “image-text” pairs (COCO dataset with translated captions)
- Visual QA – 85 759 “question-image” pairs
- TRitW – START dataset (40 312 real-images samples)

All training datasets are extended with the natural language instructions.

Tasks' Weights during Fine-Tuning

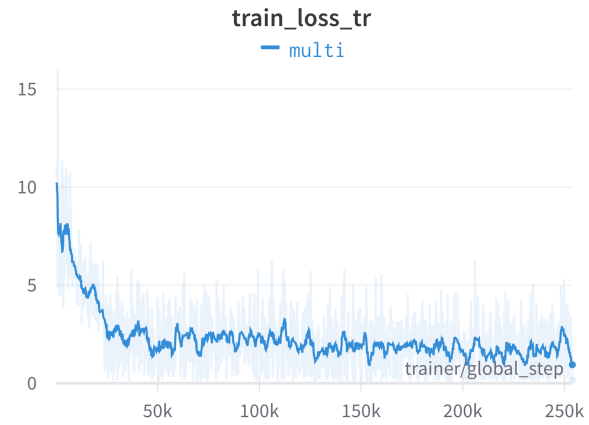
During fine-tuning you can manage the influence of each training task by changing its loss weight.

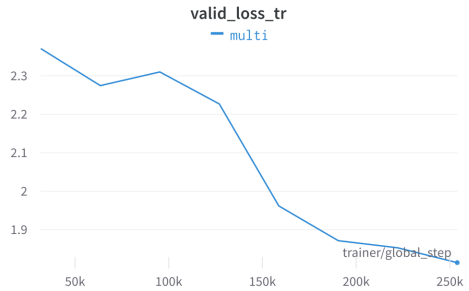
The baseline was fine-tuned with the equal (*0.5*) weight for each task.

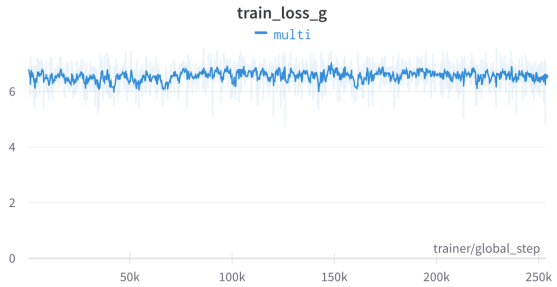
```yaml
trainer:
  bs: 2
  task_weights:
    text_loss_weight: 0.5
    math_loss_weight: 0.5
    gener_loss_weight: 0.5
    capt_loss_weight: 0.5
    vqa_loss_weight: 0.5
    text_recog_loss_weight: 0.5
```

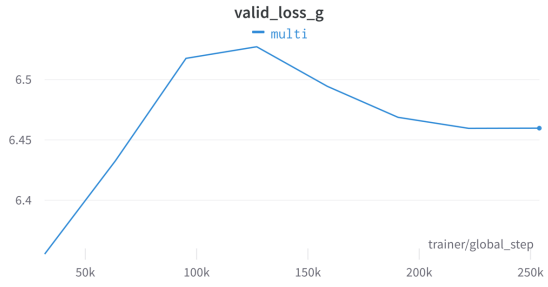

*config file on github

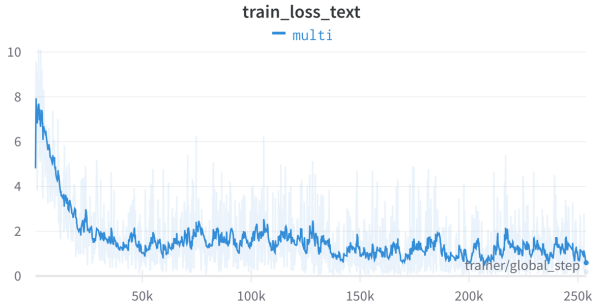
RUDOLPH Fine-Tuning

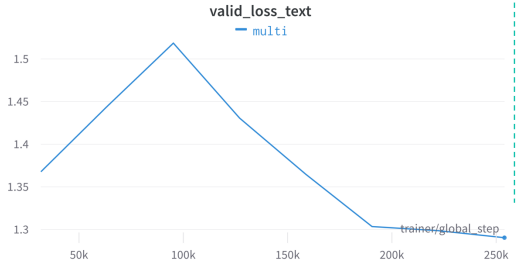


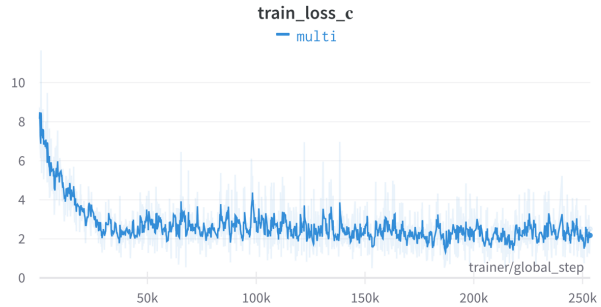


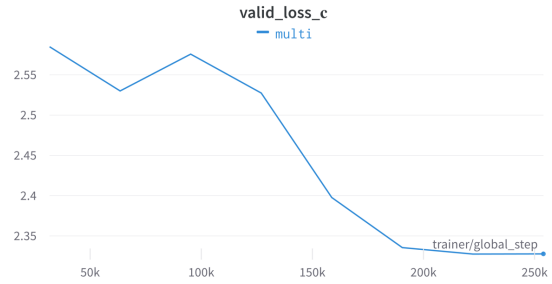












Fine-Tuned RUDOLPH Performance

	TextQA	MathQA	Image Generation	Image Captioning	Visual QA	TRitW
FBC2Authors	0.25	0.31	0.28	0.23	0.34	0.36

Hidden1	Hidden2	Hidden3	Hidden4	Hidden5	Hidden6
0.20	0.20	0.27	0.17	0.30	0.11

Final Score
0.24

**jupyter on single task RUDOLPH fine-tuning is on github*