Human perceiving in code search and generation

RLHF for Community Q&A

MENTOR : Sergey Kovalchuk
INTERN : Alexey Gorbatovski

Problem statement

Global Project Purpose

Developing AI-powered programming assistant via RLHF model training

Benefit:

Reducing development time and solving developer's problems

Challenges:

1. Evaluating the quality of LLM responses
2. Non-representativeness of linguistic metrics in a specific domains

Project Audience

Who?

- R&D NLP Team
- Programmers
- IDE Devs.

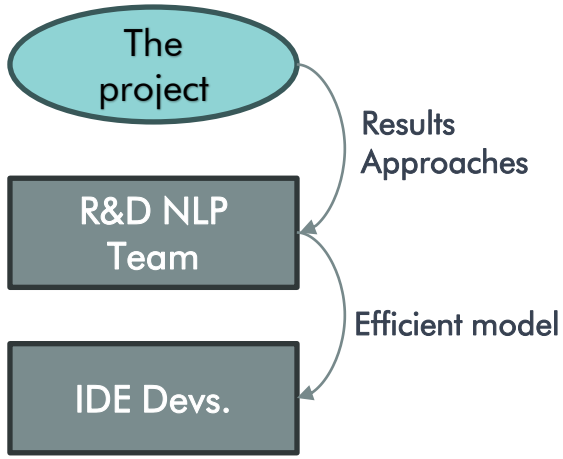
Special Focus

Developers seek AI-powered assistant tools to speed up operations

Why?

Our project offers an AI assistant capable of providing professional consultations, enriching the software development process

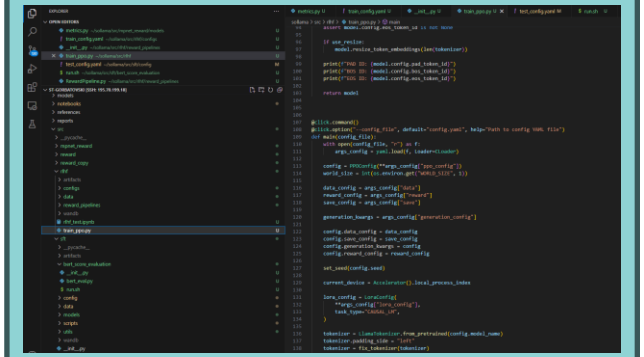
My contribution



Usage Scenarios

Primary Use Case

IDE



AI assistant

Secondary Use Case

Proactive AI tracking the developer

Accelerating and simplifying the development process

Solution Value

Innovation

Introducing an AI assistant with professional consultation capabilities, improving upon existing solutions like GitHub's Co-pilot

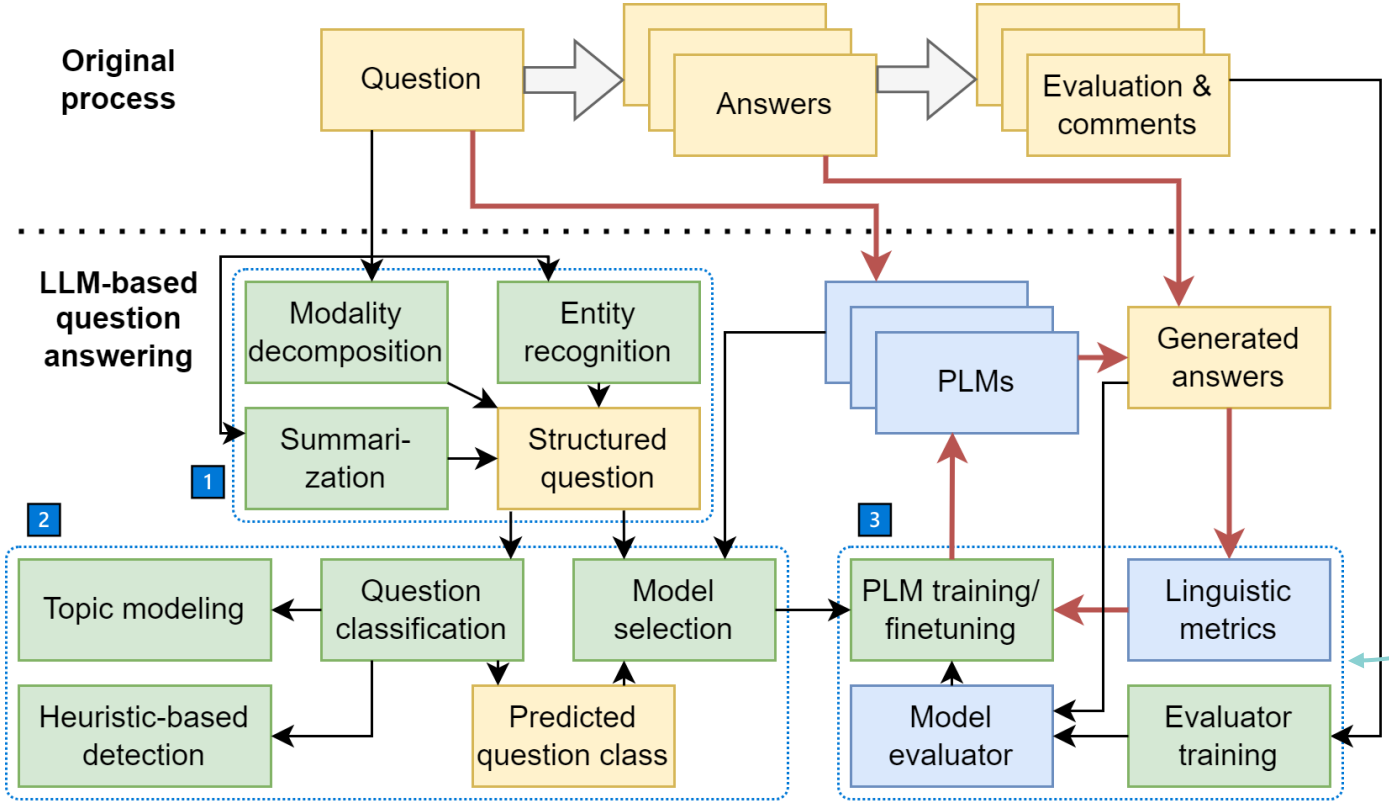
Usability

Solution can be directly integrated into IDEs, bringing intelligent assistance to the development process

Impact

Solution empowers developers to produce higher quality code, provides real-time insights, and saves valuable time during the software development process.

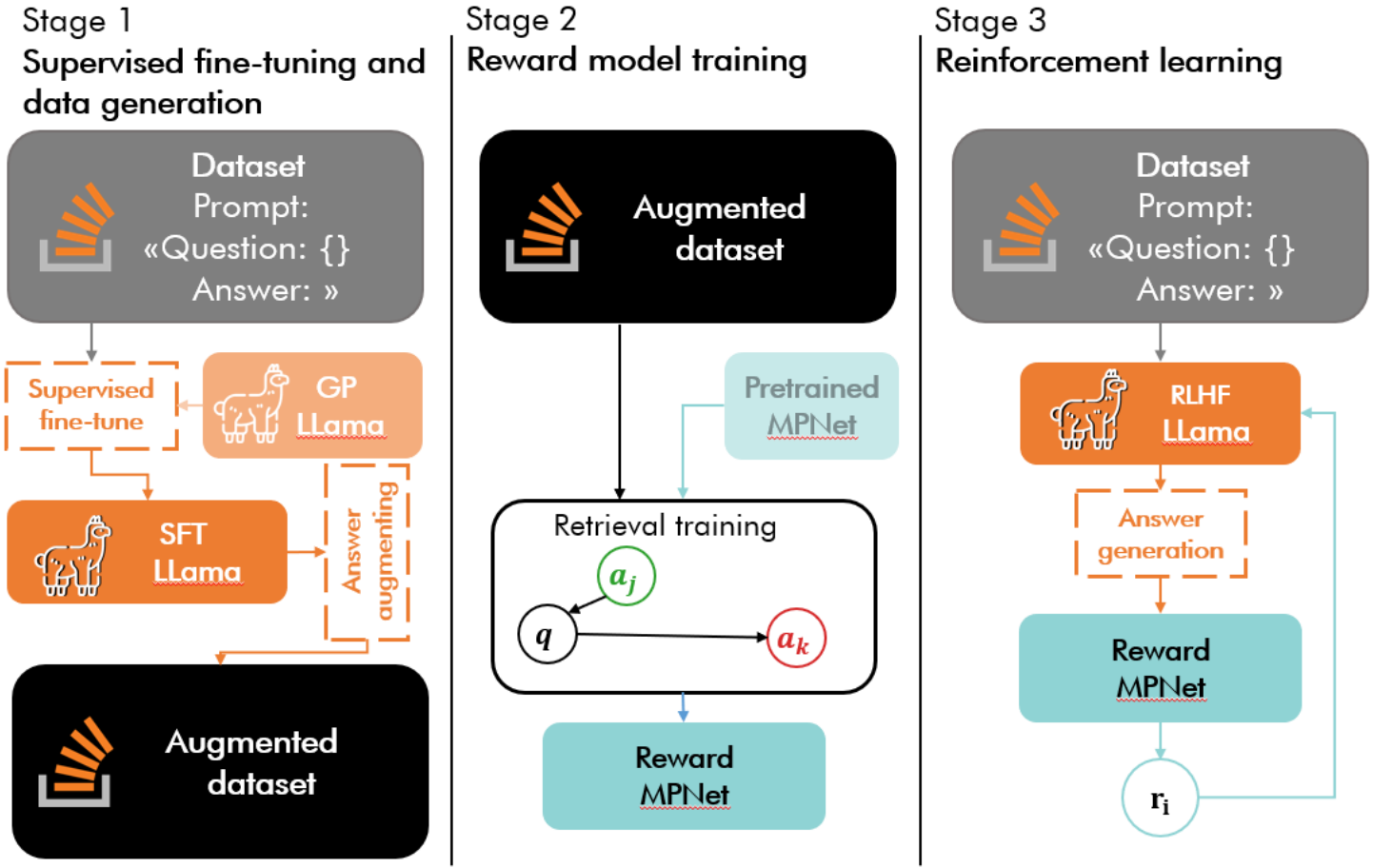
Scientific Project Purpose



Block 3

1) Model fine-tuning
2) Evaluator training

Project Scheme



Quality assurance

LLM validation

- Linguistic metrics evaluation
- Evaluation by-hands

Title question

How can I get the location of a device in python?

Generated Answer

You can look into the `geolocate` library. It has a really nice API and is really simple to install and use. pypi.org/project/geocoder/

Reward Model validation

- Accuracy and Pearson corr. comparing
- Evaluation by-hands
- Ranking comparing w/ linguistic metrics
- RL LLM training results

Limitations

- Stack Overflow Q&A
- *Python domain*
- *Python basics category*

Reward Model Results

Retrieval reward training approaches

- Dot product similarity
  - Average accuracy: 95.77%
  - Correlation with log score: 0.36
- Cosine distance
  - Average accuracy: 98.16%
  - Correlation with log score: 0.35

Comparison with linguistic metrics

- ROUGE 1 – 0.15
- ROUGE 2 – 0.1
- ROUGE L – 0.11

** growth from 0.01 up to current values*

Paraphrasing
Accelerate RM accuracy
74% vs 94%
Pretrained T5 model
Ref:humarin/chatgpt_paraphraser_on_T5_base

Different triplets

Positives	Negatives	RLHF result
SO	SO	+
SO	LLM Answers	–
SO - Paraphrased	SO - Paraphrased	+

*SO - Stack Overflow

LLM Results

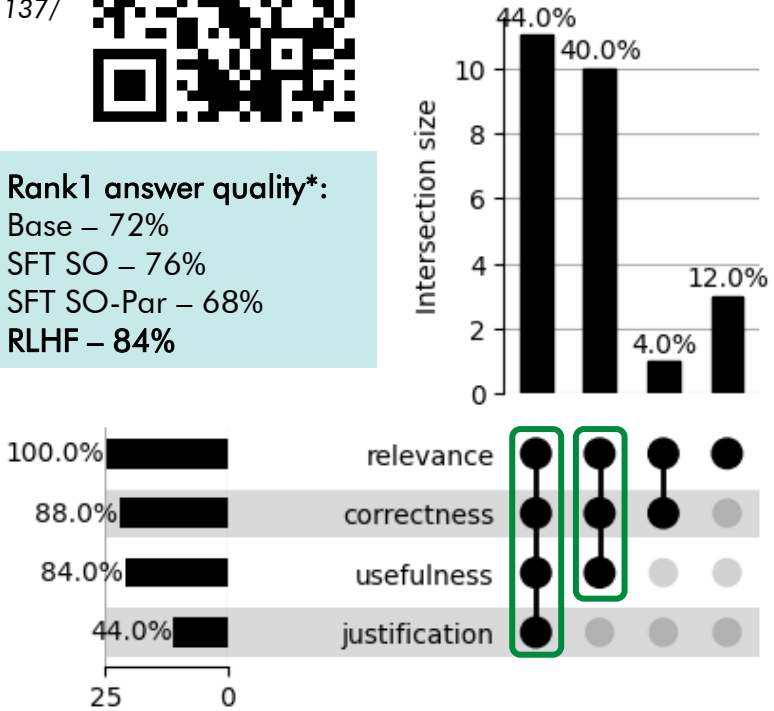
Own model validation form
http://45.87.153.137/

Python Basics category - 3 attempts

Model/ Metric	GP LLaMA 7B	SFT LoRA LLaMA	SFT LoRA LLaMA w/ Par	RLHF LLaMA SO-Paraphrased
ROUGE 1	0.1889 (σ : 0.0676)	0.1942 (σ : 0.0632)	0.1937 (σ : 0.0632)	**0.1978** (σ : 0.0639)
ROUGE 2	0.0218 (σ : 0.0173)	0.0228 (σ : 0.0206)	0.0207 (σ : 0.0183)	**0.0229** (σ : 0.0201)
ROUGE L	0.1143 (σ : 0.0339)	0.1247 (σ : 0.0365)	0.1232 (σ : 0.0357)	**0.1271** (σ : 0.0375)
SacreBLEU	0.0472 (σ : 0.0143)	0.0500 (σ : 0.0176)	**0.0504** (σ : 0.0149)	0.0500 (σ : 0.0236)
Bert Score	0.9441 (σ : 0.0073)	0.9480 (σ : 0.0076)	**0.9490** (σ : 0.0071)	0.9470 (σ : 0.0065)
Tokens	128.5356 (σ : 50.81)	66.6903 (σ : 29.08)	58.3351 (σ : 21.79)	57.1811 (σ : 11.99)

Rank1 answer quality*:
Base – 72%
SFT SO – 76%
SFT SO-Par – 68%
RLHF – 84%

RLHF UpSet Plot



*correct&useful&relevant

Release 🤗

Method

Paper: «Community question answering in complex domains with pretrained language models: application to programming domain»

Code

GitHub repository:
https://github.com/Myashka/sollama

Model *(weights)*

RLHF LLM Hugging Face repository:
https://huggingface.co/Myashka/LLama_RLHF_SO

Reward Model Hugging Face repository:
https://huggingface.co/Myashka/MPNet_RM

Dataset

Hugging Face repository:
huggingface.co/Myashka/SO_Python_basics_QA_human_pref

Alexey Gorbatovski
Intern Cloud BU
gorbatovski@itmo.ru

Thank you for your attention!

Datasets

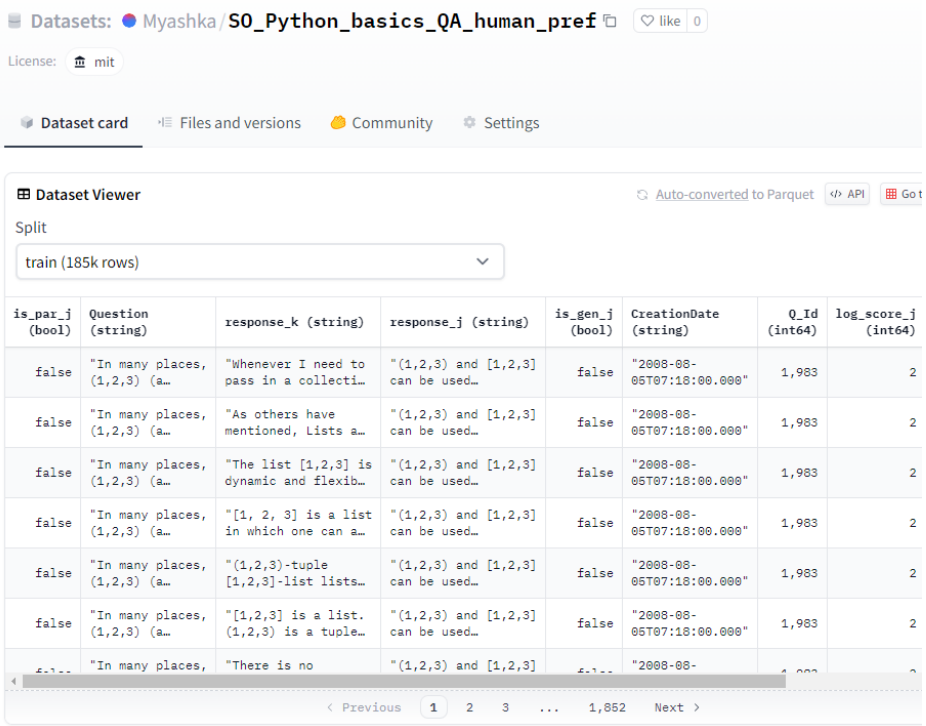
Stack Overflow Python basics dataset

Ref: Myashka/SO_Python_basics_QA-filtered-2023-T5_paraphrased-tanh_score

- Python basics category
- No code blocks
- No images
- No links

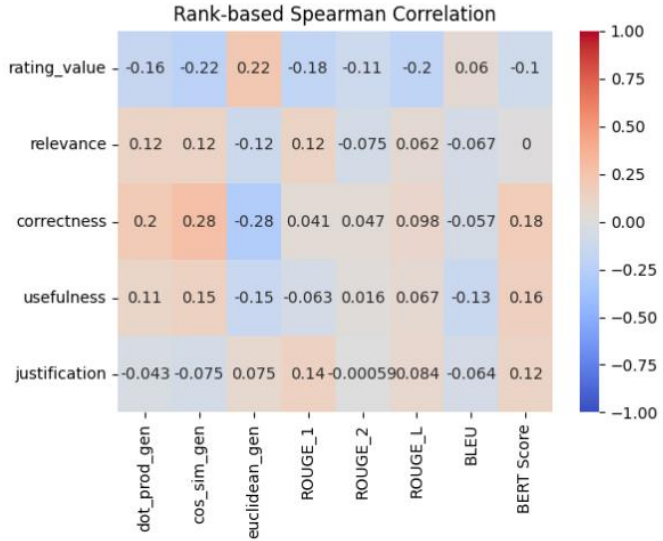
Human preference dataset

Ref: Myashka/SO_Python_basics_QA_human_pref

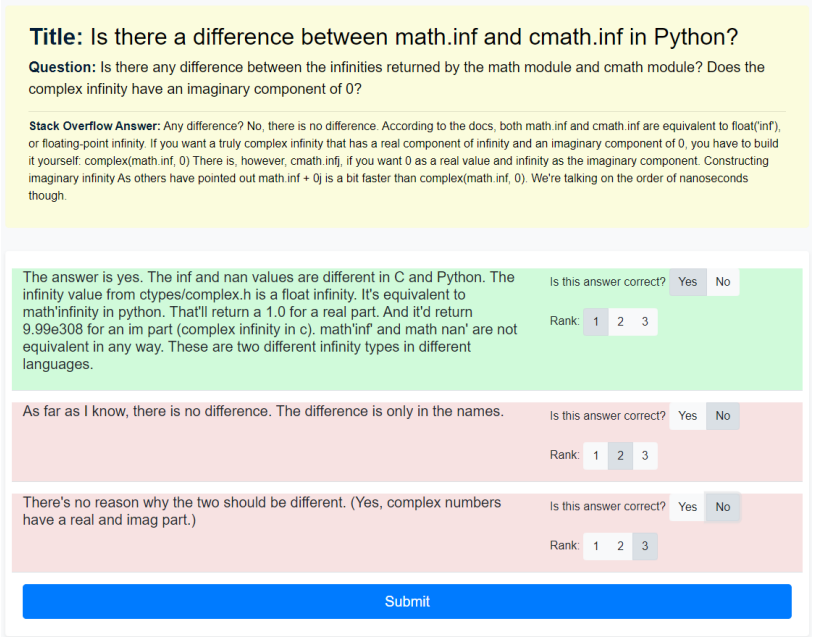


Human Evaluation

RLHF Correlations aka Metrics ambiguity



Own model validation form
http://45.87.153.137/



Reward Model Training

- **Model**

MPNet

Ref: sentence-transformers/multi-qa-mpnet-base-cos-v1

- **Objective**

Triplet loss

$$Loss = \sum_{i=1}^{N} \left[\|f_i^a - f_i^p\|_2^2 - \|f_i^a - f_i^n\|_2^2 + \alpha \right]_+$$

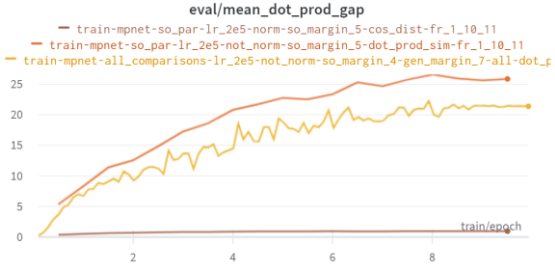
Paraphrasing

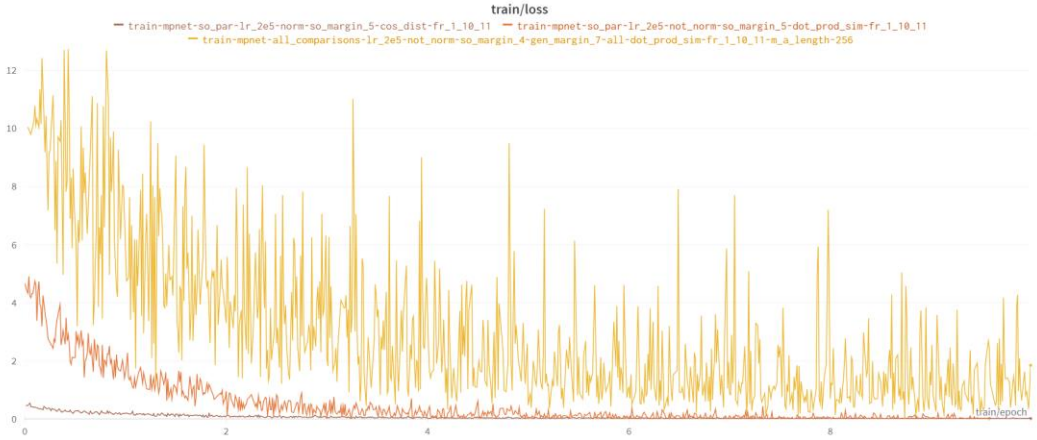
Help RM training

74% vs 94%

T5 model

Ref:humarin/chatgpt_paraphraser_on_T5_base





Used measures

- Dot product
- Cosine similarity

PPO LLaMA Training

Rewards

- Cosine
- Dot product

Conclusions

1) Normalized reward more stable
2) Dot product reward degenerates to «Yes»/«No» answers
3) Length penalty may be good and bad as well