AI Explainability WG

March 2024

March 2024 updates

- TrustyAI core / service
    - 0.11.1 release
    - https://github.com/trustyai-explainability/trustyai-explainability/releases/tag/v0.11.1
    - quay.io/trustyai/trustyai-service:v0.11.1
- TrustyAI operator
    - 1.17.0 release
    - https://github.com/trustyai-explainability/trustyai-service-operator/releases/tag/v1.17.0
    - quay.io/trustyai/trustyai-service-operator:v1.17.0
- TrustyAI Python
    - 0.6.0 release
    - https://github.com/trustyai-explainability/trustyai-explainability-python/releases/tag/0.6.0
    - https://pypi.org/project/trustyai/0.6.0/

What’s new?

TrustyAI - What’s new?

- **TrustyAI core/service 0.11.1**
  - Fix single-valued metrics being wrapped into arrays in request listings
  - Update Quarkus from 3.2.7.Final to 3.2.11.Final
  - [CI] Migration to ODHv2, numerous improvements
- **TrustyAI operator 1.17.0**
  - Change oauth-proxy image to Quay's for community releases
  - Refactor operator's ServiceMonitors, Service and Route into templates
  - Add ODH Trusted CA bundle support
  - Fix coordination and leases permissions
  - [CI] Add Kind-based smoke tests to PRs
  - [CVE] CVE-2023-48795
- **Python TrustyAI 0.6.0**
  - TrustyAI service integration[1]
    - Add methods to execute and return queries from a service
  - Decouple visualisation from core explainability

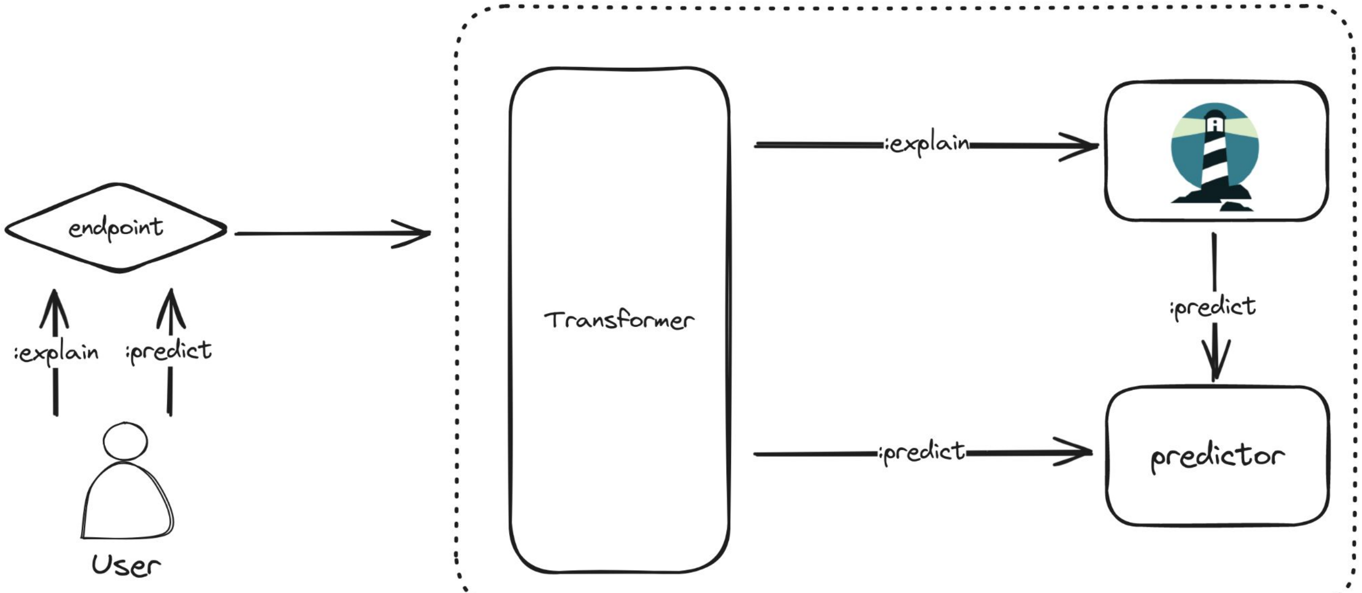
[1]: https://github.com/trustyai-explainability/trustyai-explainability-python-examples/blob/main/examples/DataDrift.ipynb

Current work

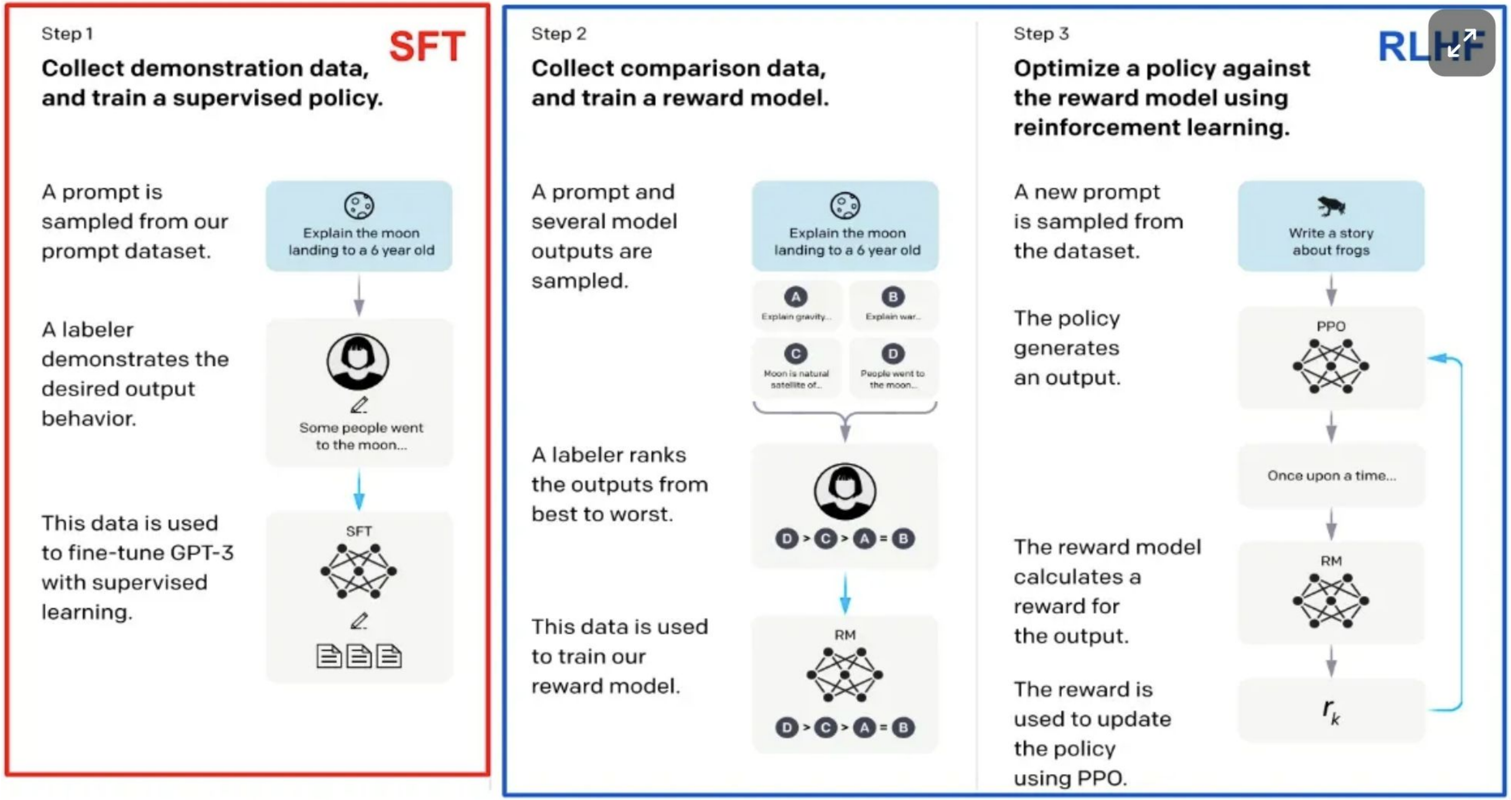
TrustyAI - current work

- **TrustyAI core/service**
  - Support for database storage
  - Fixing Explainer issues (Service)
    - Tensor parsing
    - Better gRPC resource management
    - Exclude synthetic payloads from bias metrics
  - Language metrics
    - ROUGE
- **TrustyAI operator**
  - TLS-enabled Kubernetes `Service` for payload consumer
  - Fixes to TrustyAIService finalizer

KServe explainer



Detoxify LLMs through Reinforcement Learning (RL)



Step 1

Collect demonstration data, and train a supervised policy.

A prompt is sampled from our prompt dataset.

A labeler demonstrates the desired output behavior.

This data is used to fine-tune GPT-3 with supervised learning.

Step 2

Collect comparison data, and train a reward model.

A prompt and several model outputs are sampled.

A labeler ranks the outputs from best to worst.

This data is used to train our reward model.

Step 3

Optimize a policy against the reward model using reinforcement learning.

A new prompt is sampled from the dataset.

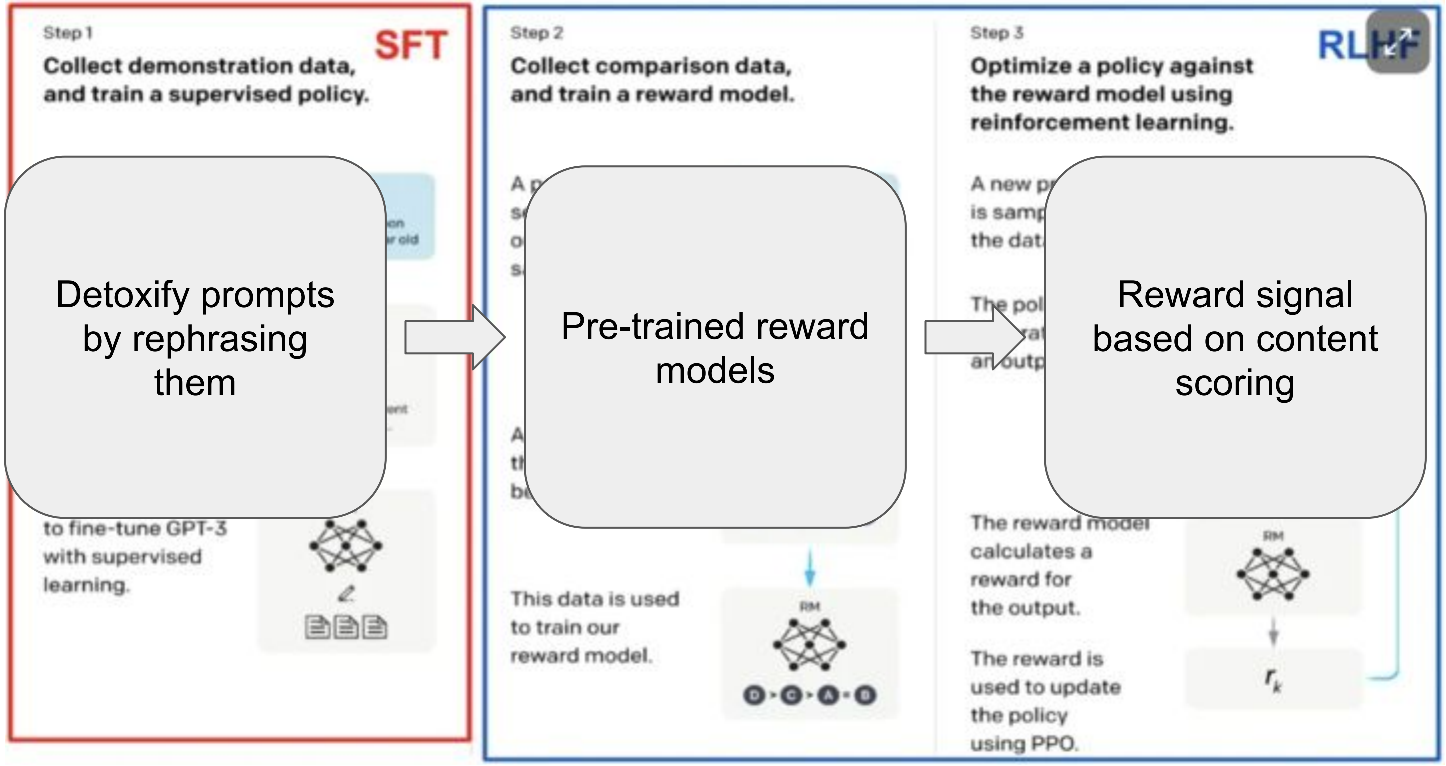
The policy generates an output.

The reward model calculates a reward for the output.

The reward is used to update the policy using PPO.

SFT
Explain the moon landing to a 6 year old
Some people went to the moon...
SFT
A
Explain gravity...
B
Explain war...
C
Moon is natural satellite of...
D
People went to the moon...
D > C > A = B
RM
D > C > A = B
RLHF
Write a story about frogs
PPO
Once upon a time...
RM
r_k

How TrustyAI Detoxify can streamline RL



Step 1

Collect demonstration data, and train a supervised policy.

to fine-tune GPT-3 with supervised learning.

Step 2

Collect comparison data, and train a reward model.

This data is used to train our reward model.

Step 3

Optimize a policy against the reward model using reinforcement learning.

The reward model calculates a reward for the output.

The reward is used to update the policy using PPO.

Detoxify prompts by rephrasing them

Pre-trained reward models

Reward signal based on content scoring

Community

TrustyAI - community

- **GitHub discussions**
  - https://github.com/orgs/trustyai-explainability/discussions

Roadmap

TrustyAI 2024 roadmap

- **December 2023 – March 2024 (proposal / discussion)**
  - Out-of-distribution (OOD) metrics completion
    - Finalise OOD metrics, namely data uploading aspect
      - Provide data connection for data upload
  - Explainability service endpoints completion
    - Formalise explainability payloads schema
    - Refining handling of synthetic payloads to avoid interference with metrics
    - Handle potential large computational times in a service setting
  - Detoxification at the library and service level
    - Integrate detoxification with Python TrustyAI (Jupyter main target)
    - Token scoring at the service level
  - Database backend
    - Address scalability
    - Replace PVC with DB?
  - Expand supported types (*eg* image data)
    - Metrics and explainability for non-tabular data
  - Model drift/data drift/anomaly detection
  - Improve handling of unsupported model serving runtimes

Legend

Not started

In progress

Completed

TrustyAI 2024 roadmap

- **March 2024 - May 2024 (proposal / discussion)**
  - KServe explainer integration
  - Detoxification fine-tuning

Legend

Not started

In progress

Completed