Responsible AI Explain APIs

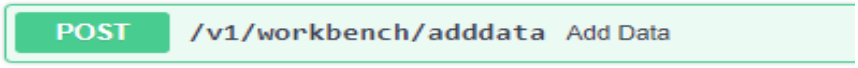
Please follow the below instructions to use Explainable AI APIs.

First, you have to upload the Model, Dataset & Preprocessor (if applicable) to the MongoDB using Workbench APIs from **responsible-ai-model-details** repository as detailed below:

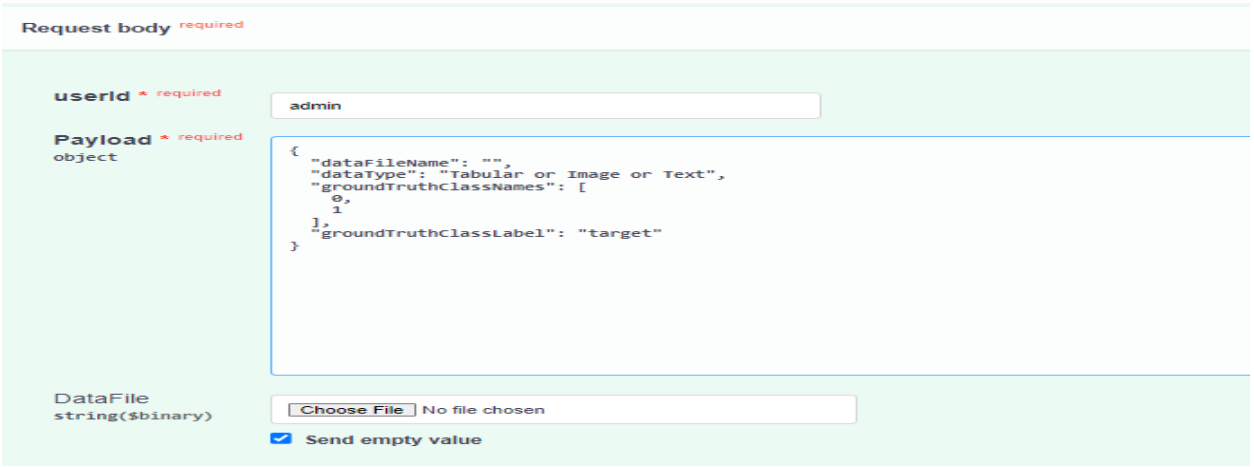
Step-1:

Upload Dataset:

Please go to the "adddata" API at the URL mentioned above.



Attached is the requested payload below.



userId : The "userId" parameter, which corresponds to the "usecase_name."

Payload:

dataFileName: The "dataFileName" field should contain the name of your data file

dataType: The "dataType" field should specify the type of data, such as "tabular" or "text," based on the dataset.

groundTruthClassNames: "groundTruthClassNames" typically refers to the classes or categories present in a dataset, particularly in supervised learning tasks where the data is labeled. For example, in a dataset of images of fruits categorized into apples, bananas, and oranges, "apple," "banana," and "orange" would be the ground truth class names. This information helps in training machine learning models to correctly classify or predict the data. If the dataset contains class labels, they would be listed in the "groundTruthClassNames" field. If there are no class labels or categories in the dataset, this field would be left empty.

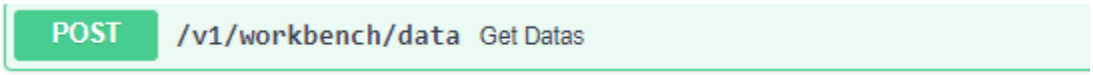
groundTruthClassLabel: Mention the target column name that your model is going to predict.

DataFile: Please select and upload the dataset file by browsing your device.

After entering all the necessary data, proceed by clicking on "execute."

If the information is successfully saved in the database, you will receive a response stating "Data added successfully.

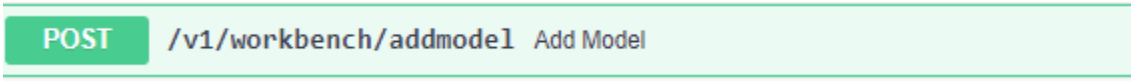
To retrieve the details of the uploaded data, please navigate to the following API. You will receive information such as dataId and name etc.



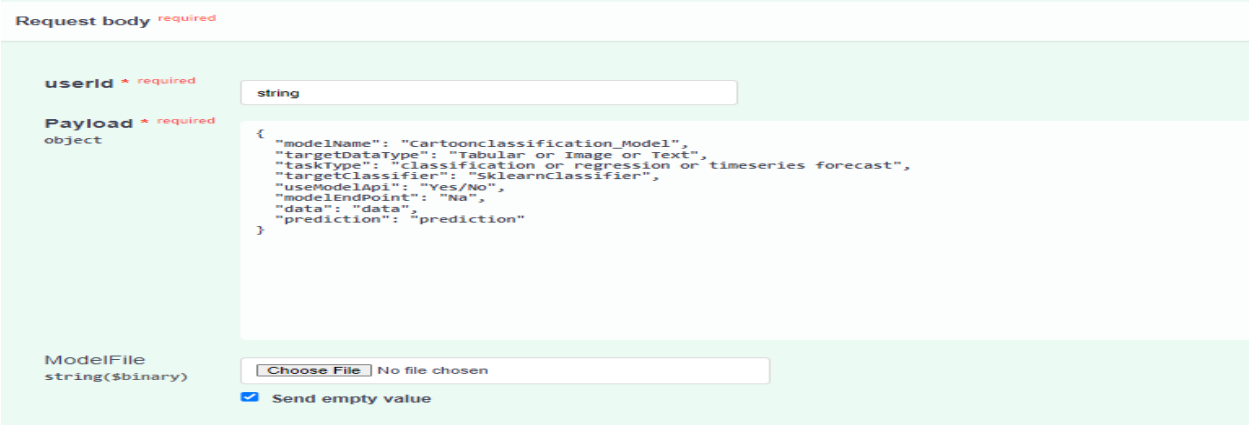
Step-2:

Model Upload:

Please access the "addmodel" API at the provided URL.



The requested payload is attached below



userId : The "userId" parameter, which corresponds to the "user_name."

Payload:

modelName : The "modelName" field should contain the name you choose for your model.

targetDataType: The "targetDataType" field should specify the type of data your model is designed to work with, such as "tabular" or "text," based on the dataset it will be trained on

taskType: The "taskType" field should specify the type of task your model is intended for. You can provide "CLASSIFICATION" if the model is for classification tasks, "REGRESSION" if it's for regression tasks, or "TIMESERIESFORECAST" if it's for time series forecasting

targetClassifier: You can remove or ignore the "targetClassifier" field.

useModelApi: If you are uploading the model, please provide "**No**." Otherwise, if you are providing the model via an endpoint, provide "**Yes**."

If "**useModelApi**" is set to "**Yes**":

modelEndpoint: If you are accessing the model via an endpoint, please specify your endpoint here in the "modelEndpoint" field.

data: If you are accessing the model via an endpoint, the "data" field should contain the input parameter of the endpoint, which binds input data to the endpoint.

prediction: If you are accessing the model via an endpoint, the "prediction" field should contain the output parameter of the endpoint, which delivers data from the endpoint.

ModelFile: Please ignore this field.

If "**useModelApi**" is set to "**No**":

In this case, you can either remove these three fields or leave them as they are:

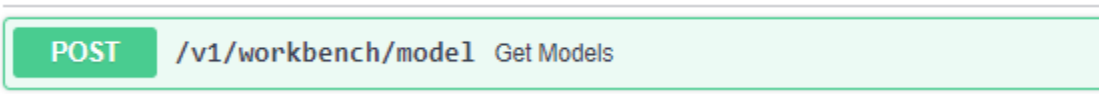
- modelEndpoint
- data
- prediction

ModelFile: Please select and upload the model file by browsing your device.

After entering all the necessary data, click on "execute."

If the model is successfully saved in the database, you will receive a response stating "Model added successfully."

To retrieve the details of the uploaded model, please navigate to the following API. You will receive information such as modelId and name etc.

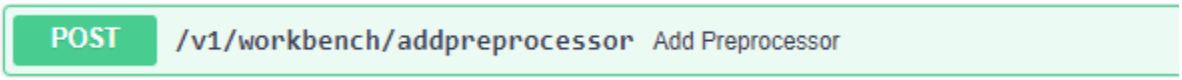


If you have a preprocessor file, please follow these steps. If you don't have one, you can skip step-3 and proceed to step-4.

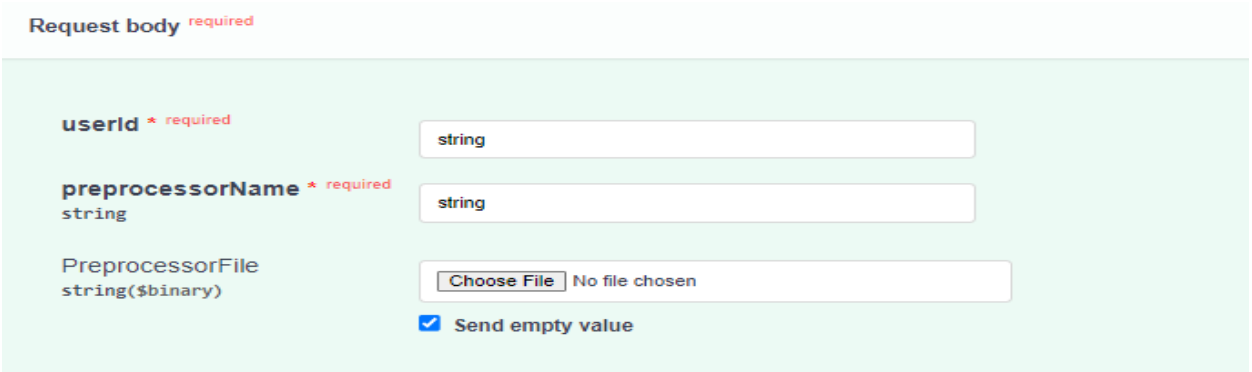
Step-3:

Preprocessor Upload:

Please access the "addpreprocessor" API at the provided URL.



The requested payload is attached below



userId : The "userId" parameter, which corresponds to the "user_name."

preprocessorName: Please provide the name of your preprocessor in the "preprocessorName" field.

PreprocessorFile: Please select and upload the preprocessor file by browsing your device.

After providing all the required data, click on "execute."

If the preprocessor is successfully saved in the database, you will receive a response stating "Preprocessor added successfully."

To retrieve the details of the uploaded preprocessor, please navigate to the following API. You will receive information such as preprocessorId and name.

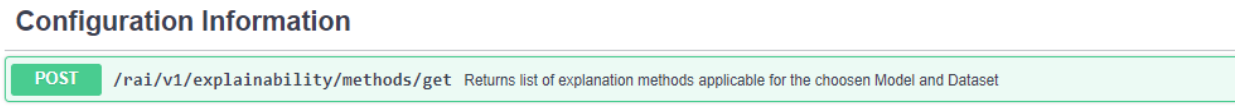
Step-4:

Batch Generation:

Now that we've uploaded all the necessary files, we can proceed to create a batch for the specified model and dataset.

Before creating a batch, we need to obtain the applicable methods. These methods can be retrieved from the "get methods" API in the Explainability docs.

Please navigate to the method mentioned in the URL above



The requested payload is attached below.

modelId: You will obtain the "modelId" from the "get models" API. (/v1/workbench/model).
datasetId: You will obtain the "dataId" from the "get datas" API. (/v1/workbench/data).

scope: Please specify "LOCAL" if you want local applicable methods and "GLOBAL" for global applicable methods. If you want all methods irrespective of scope, remove this field or pass it as null.

You will receive the applicable methods as a response.

Response body


```
{
  "status": "SUCCESS",
  "message": "Identification of explanation methods successful",
  "dataType": "Tabular",
  "methods": [
    "ANCHOR-TABULAR",
    "KERNEL-SHAP",
    "PD-VARIANCE"
  ]
}
```


Please copy those methods and paste them into the "appExplanationMethods" column in below batch generation API .

Please access the "batchgeneration" method at the provided URL.

POST /v1/workbench/batchgeneration Getbatch

Below attaching the requested payload

Request body required

Example Value | Schema


```
{
  "userId": "admin",
  "title": "Preprocessor1",
  "modelId": 1.1,
  "dataId": 2.1,
  "tenetName": [
    "string"
  ],
  "appAttacks": [
    "string"
  ],
  "appExplanationMethods": [
    "string"
  ],
  "biasType": "string",
  "methodType": "string",
  "taskType": "string",
  "label": "string",
  "favorableOutcome": "string",
  "protectedAttribute": "string",
  "privilegedGroup": "string",
  "preProcessorId": 0
}
```

userId : The "userId" parameter, which corresponds to the "user_name."

title: You can provide a suitable title for the task you are performing using the given model and dataset.

modelId: You will obtain the "modelId" from the "get models" API. (/v1/workbench/model).

dataId: You will obtain the "dataId" from the "get datas" API. (/v1/workbench/data).
tenetName: "Explainability" is the tenet name.
appAttacks: Please remove this field or leave it as it is.
appExplanationMethods: Please paste the methods you obtained from the "get methods" API in Explainability docs.
You can either remove these fields or leave them as they are:

- biasType
- methodType
- taskType
- label
- favorableOutcome
- protectedAttribute
- privilegedGroup

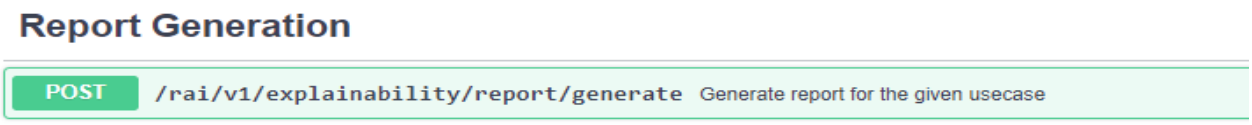
preProcessorId : For "preProcessorId," you will obtain the preprocessorId from the "get preprocessor" API. (/v1/workbench/preprocessor).

Once you have provided all the necessary data, click on "execute." You will receive the batchId and tenetId as a response.

Copy the batchId and paste it into the below API in the Explainability docs to generate the report for the given batchId.

Step-5:

Report Generation:



We need to pass the batchId as a request, and It will create the report and store it in PDF format in the database.

Now that we've created the report as well. Next, we can proceed to download the report.

Step-6:

Report Download:

Go to the **responsible-ai-reporting-tool** repository and follow the instructions below

Please navigate to the method specified in the URL above.

To download the report, please refer to API provided above. Use the same batchId that was used to generate the report.