NCATS Infrastructure Proposal for the EHR DREAM Challenge: Patient Mortality Prediction

Overview

This document describes the IT infrastructure that Sage Bionetworks use to process dockerized-model submitted in DREAM Challenges. Personas and use cases relevant to the deployment, maintenance and use of this infrastructure are also provided.

Finally, a timeline is proposed for deploying the IT infrastructure required to run the EHR DREAM Challenge: Patient Mortality Prediction (a CD2H initiative) using NCATS resources.

Personas

Challenge participant (contributor)

Description: Data scientists, statisticians, or other modelers competing in a challenge (computational expertise skewed more towards scripting, analysis, visualization; less infrastructure/IT)

Actions: Designs algorithm, creates container (or specifies image), submits container to Synapse “challenge” (evaluation queue), gets notified of container validation errors or runtime errors, gets notified of model execution completion/success, views model performance metrics in Synapse leaderboards

Challenge administrator (organizers)

Description: Project leads/sponsors, research scientists (with data science and some domain expertise), and domain experts

Actions: Designs challenge questions; prepares challenge training and validation data; designs and tests baseline models; specifies submission evaluation metrics/procedures; supervises challenge operations and triages issues

Challenge IT engineer

Description: Research associates and/or data engineers, typically proficient in cloud computing and related infrastructure setup, scripting, and API/client usage

Actions: Set up, configures, maintains challenge evaluation queues; sets up and maintains cloud compute environment for evaluating submissions; documents challenge infrastructure and submission process; provides technical support for discussion forums and debugging submission/evaluation errors

Use Cases

This section describes the use cases that involve interactions with Synapse and NCATS IT infrastructure.

Use case 1: A team of IT engineers at Sage needs to deploy the challenge submission pipeline described in Section IT Infrastructure that will enable to train and evaluate the models submitted by the Challenge participants.

- **Functional needs:**
    - SSH access to an AWS EC2 instantiated by NCATS engineers using NCATS AWS account.
        - The EC2 must have as many CPU cores and memory available as possible to process models in parallel.
        - The EC2 must have access to sufficient EBS storage to host the challenge dataset.
    - The codebase of the challenge submission pipeline, which has been developed by Sage and is ready to be deployed.

Use case 2: A Challenge participant submits a docker-based model to be trained and evaluated on NCATS infrastructure.

- **Functional needs:**
    - A Docker repository where the Challenge participants can push the dockerized model to submit. Synapse is providing this repository (docker.synapse.org).
    - A way for the Challenge participant to submit a model to the challenge. The participant can achieve this using Synapse web, REST API, or R / Python clients.
    - Documentation on how to submit a model. This documentation is provided on the challenge website (the documentation is currently only available on the staging website).

Use case 3: A Challenge participant wants to track the progress of his/her submission.

- **Functional needs:**
    - The challenge agent that runs the submitted models in an NCATS EC2 must be able to communicate to Synapse the status of the submission (SUBMITTED, RUNNING, SCORED, DONE, FAILED).

    - The Challenge participant must be able to see the status of his/her submission as well as the last time the status has been updated. Synapse provides each participant with a dashboard that includes this information.

Use case 4: A challenge participant wants to get the performance of his/her model that has successfully been processed.

- **Functional needs:**
    - The challenge agent that runs the submitted models in an NCATS EC2 must be able to communicate to Synapse one or more performance metrics computed on NCATS EC2.
    - A way for the participant to get the performance of the model. Synapse provides each participant with a dashboard that includes this information. This information can also be retrieved using Synapse REST API and R / Python clients.

Use case 5: A Challenge participant wants to cancel a submission.

- **Functional needs:**
    - A way for the participant to cancel a submission. Synapse provides each participant with a dashboard that includes a button to cancel submissions that have not been completed yet.
    - If the submitted model is running, Synapse must be able to notify the challenge agent running on the NCATS EC2 that the model evaluation must be stopped. The agent then returns a confirmation to Synapse, which will then be used to notify the Challenge participant that the submission has been cancelled.

Use case 6: A team of IT engineers at Sage is informed that the submission pipeline is down, or that the status of a model has not been updated in several hours.

- **Functional needs:**
    - A team of IT engineers at Sage must be able to ssh to the NCATS EC2 instance running the challenge agent to resolve the issue. This involves identifying the cause of the issue by looking at the logs of the host/docker and attempt to restart the model(s) affected by the issue.

IT Infrastructure

Figure 1 describes the IT infrastructure that will be used for the EHR DREAM Challenge. The following sections describes how users will interact with this infrastructure as well provide specifications related to the deployment of this infrastructure.

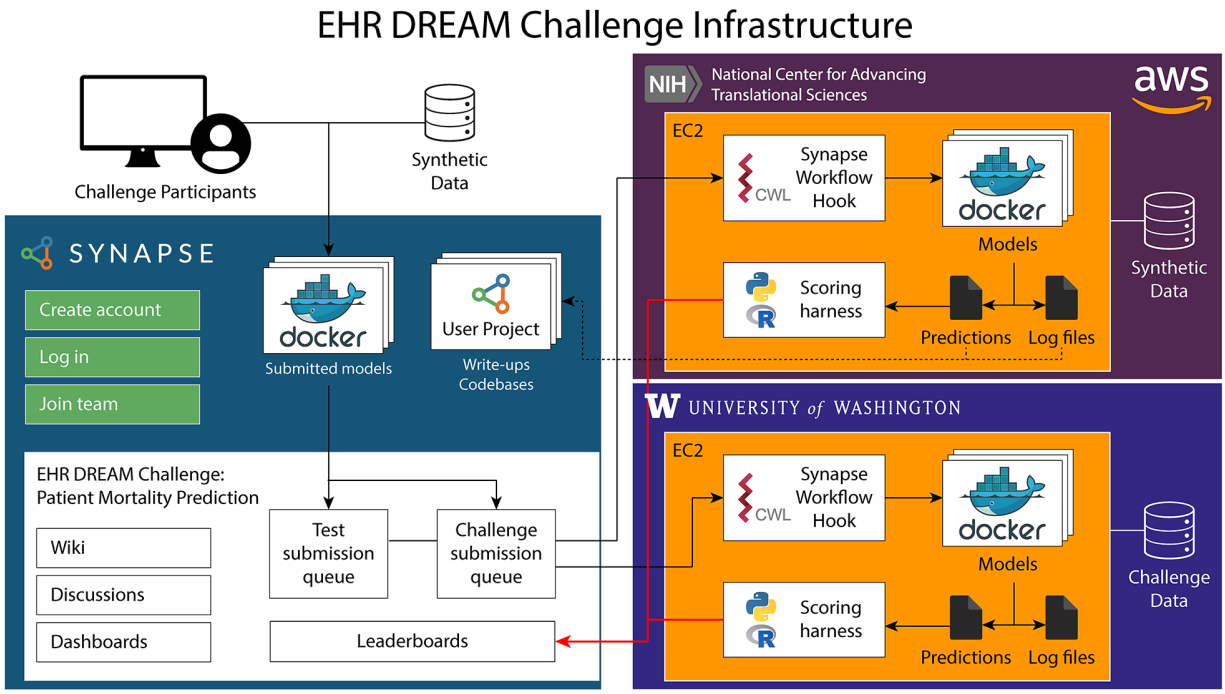

Figure 1. Cloud infrastructure that is being put in place for the EHR DREAM Challenge: Patient Mortality Predictions.

Registration to the challenge

First, challenge participants need to create an account on **Synapse**. Synapse users can participate in a challenge as an individual participant. They can also create a team or join one.

After registering to the EHR DREAM Challenge, participants will have access to documentation (**Wiki**), discussion forum and additional resources (e.g., sample data) that the challenge organizer may provide. For example, the challenge organizers will release a synthetic dataset that the participants can explore since the challenge data can not be shared with the participants.

Developing models

Using the documentation provided on the Synapse project of the EHR DREAM Challenge, participants will develop models locally — that is, using their own computing resources. Participants are also authorized to train their models on any public or private data that they have access to.

The models must then be “Dockerized” following the information provided by the challenge organizers. For example, all the models that will be submitted to this challenge must expect to read the input data files at the location specified by the organizers. The models are also required to output their predictions at a specific location and in the format specified by the challenge organizers. Moreover, some information relevant to the task performed by the models

may be provided as environment variables in order to make the training and evaluations of the more reusable to address slightly different scientific questions (e.g., predict the death status of a patient in 3 months, 6 months, etc.). The documentation related to these different environment variables will be available on the challenge website.

Submitting Dockerized models

Synapse offers a Docker repository for each project created. Challenge participants can then push their dockerized models to their own Synapse project or to the project shared with their other teammates. These models are then visible under the Docker tab of their project. From there, a model can be easily submitted to one of the challenge submission queues (**Fig. 2**). Different submission queues are usually created when the challenge targets to address more than one scientific questions or run models on different datasets.

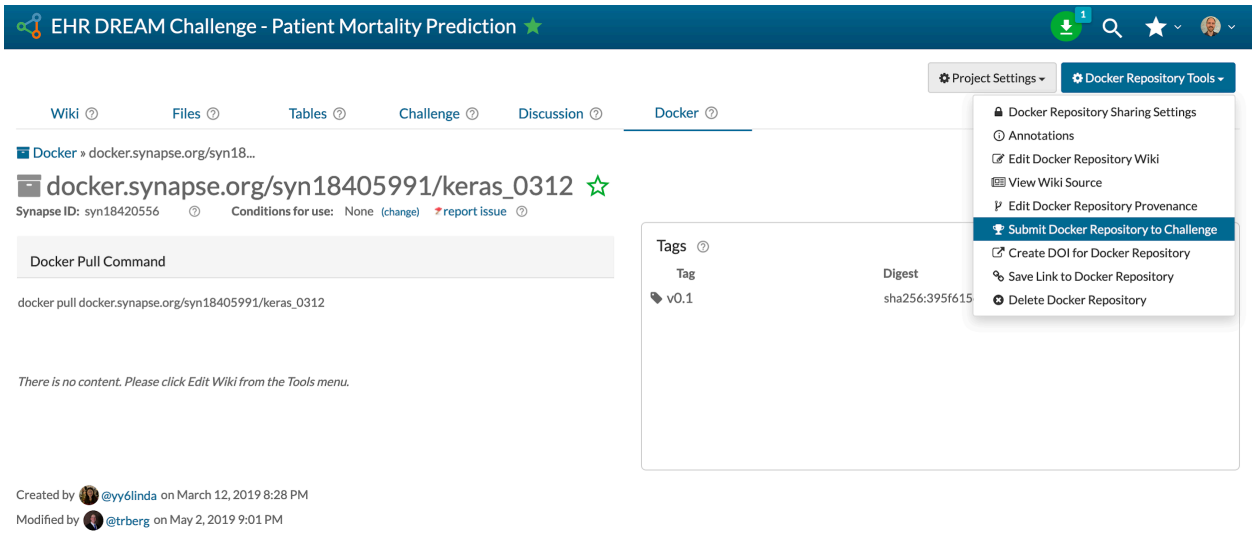

Figure 2. Challenge participants can push their model to the Synapse Docker registry. Using Synapse, participants can easily submit a dockerized models to one of the challenge submission queues.

Running models in a secure environment

Sage Bionetworks has developed a [Toil](#)-based **agent** called the [Synapse Workflow Hook](#). This agent is typically deployed on an AWS EC2 instance and pull submissions made to one or more challenge submissions queues (see **Fig. 1, bottom-right**). The agent is configured using a [CWL (Common Workflow Language)](#) file that describes all the steps that must be applied when processing a model. For example, one step could be to perform some tests on the submitted models to ensure that it meets some requirements defined by the challenge organizers.

Thanks to the Toil workflow execution engine used by the agent, multiple submissions can be processed at the same time if the EC2 running the agent has enough computational resources

for running *N* submissions in parallel, that is, enough number of CPU cores, amount of memory, disk space to save intermediary and output files, GPUs, etc.

Note: The horizontal deployments of the agent, that is the ability to spin up additional EC2 instances to process more models in parallel, is currently under development at Sage Bionetworks. Instead, a sufficiently large EC2 must be instantiated in order to run as many submissions as possible in parallel.

When running a model, the model is given access to the dataset files with Read Only permission. A model also has its network interface disable, which **prevents it to upload files outside of the secure environment**.

While the model is running, the agent can be configured to upload the model's log file to Synapse, which can then be shared automatically with the participant or team who submitted the model. This feature is useful to enable participants to troubleshoot errors that their model may encounter. Returning the log file of a model would usually only be returned when the dataset exposed to the models can be shared with the participants, for example when the agent is deployment to process models submitted to a test submission queue that uses synthetic or de-identified dataset (see **Fig. 1, top-right**)

Evaluating the performance of models

Upon successful completion of the model, the predictions generated by the model is then scored using a Scoring harness coded in R or Python. This scoring harness takes as input a predictions file and the gold standard file (also known as Ground Truth), and computed one or more performance metrics identified by the challenge organizers.

Returning results to the participants and leaderboards

The performance metrics computed are then pushed to Synapse and shared by email with the participants or teams who made the submission. In addition, these metrics can be automatically added to leaderboard tables on the challenge website. In case of a live leaderboard, the performance of a model is automatically added to a leaderboard table so that any participant can see the results of the other participants.

Submission Dashboards

Participants can visualize the status (queued, running, error, success, scored) of their submission using a dashboard only visible them or their team on the challenge website. Participants can also cancel submissions by clicking on a button associated to a submission.

Running models on the UW data

Due to the fact that the data cannot be shared with the participants, neither the log files or predictions files will be returned to the participants via Synapse (see **Fig. 1, bottom-right**). The only piece of information that will leave the UW infrastructure is the performance metrics that will be pushed to Synapse.

Running models on NCATS

To support the EHR DREAM Challenge: Patient Mortality Prediction, the idea is to deploy the above submission pipeline (see **Fig. 1**) on NCATS infrastructure. For this challenge, the idea is to use NCATS infrastructure to provide challenge participants with a test submission queue that will test the submitted model on a synthetic dataset. The dataset that will be used will be generated from the **SynPUF dataset**. Because this dataset does not include any PHI information, the log file and predictions file generated by the model will be shared with the individual participant or team that submitted the model. Using this information, participants will be able to submit more robust models to challenge submission queue that process models on the UW data.

AWS Resources required to deploy the challenge submission pipeline

- The largest EC2 that UW/NCATS can provide in terms of CPU cores and amount of memory in order to enable a large number of models to be processed in parallel.
- Two instances of the selected EC2: the first for processing models submitted to the test submission queue and running on synthetic data, the second to run models on the challenge data.
- Once the agents have been deployed, all the ports of the EC2 can be closed. The only requirement is that the agent must be able to communicate with synapse.org in order to pull models from the submission queues and post performance metrics back to Synapse.

Synapse Workflow Hook codebase

The codebase of the Synapse Workflow Hook agent that would be installed on NCATS infrastructure is publicly available here

https://github.com/Sage-Bionetworks/SynapseWorkflowHook

Note: This GitHub repository also includes documentation on how to setup the agent. However, the setup should ideally be deployed by or with the assistance of a Sage engineer.

The actions performed by the agent to process a submission are described in a CWL template. This template is usually challenge specific to address their different needs (e.g., specific tests applied to validate the model before running it on the data).

Example of EHR model

This section describes how to run locally the current version of the baseline model (https://github.com/yy6linda/mortality_prediction_docker_model/) that will be used in the EHR DREAM Challenge: Patient Mortality Prediction. The following instructions can be used to train and generate predictions on a small subset of the synthetic dataset SynPUF. Here the goal is to illustrate how a submitted model will be run. While the instructions below enable to run a model manually, this action will be performed by the Synapse Workflow Hook agent (see **Fig. 1**).

A Synapse account is required to access the resources listed below. Please share your Synapse username with thomas.schaffter@sagebase.org in order to get access to the following resources.

1. Download a subset of the synthetic SynPUF dataset here.
2. Uncompress the archive in the current directory

```
tar xjvf synpuf_clean_20190627.tar.bz2
```

3. Login to Synapse Docker registry using your Synapse user credentials:

```
docker login docker.synapse.org
```

4. Train the baseline model:

```
export EHR_PATH=$(pwd) && \
  mkdir -p scratch model output && \
  docker run --network=none \
  -v $EHR_PATH/synpuf_clean/train:/train:ro \
  -v $EHR_PATH/scratch:/scratch:rw \
  -v $EHR_PATH/model:/model:rw \
  docker.synapse.org/syn18405992/0626_demographic:v0.1 bash /app/train.sh
```

5. Generate mortality predictions using the trained model (model/baseline.joblib):

```
export EHR_PATH=$(pwd) && \
  mkdir -p scratch model output && \
  docker run --network=none \
  -v $EHR_PATH/synpuf_clean/validation:/infer:ro \
  -v $EHR_PATH/scratch:/scratch:rw \
  -v $EHR_PATH/model:/model:ro \
  -v $EHR_PATH/output:/output:rw \
  docker.synapse.org/syn18405992/0626_demographic:v0.1 bash /app/infer.sh
```

The predictions file generated is available at `output/predictions.csv`.

Timeline

Here is a suggested timeline for 1) instantiating the required NCATS resources, 2) deployment and testing of the challenge submission pipeline described in Section IT Infrastructure, and 3) start accepting and processing dockerized models submitted by the Challenge participants.

By July 19:

- NCATS IT Engineers will instantiate an EC2 for the challenge
- NCATS IT Engineers will grant access to the EC2 to the team of IT Engineers at Sage Bionetworks

By July 31:

- The team of IT Engineers at Sage Bionetworks will deploy and test the challenge agent (see **Fig. 1**) on the NCATS EC2.

Mid-August:

- Challenge participants starts submitting models to run on NCATS EC2 instance.