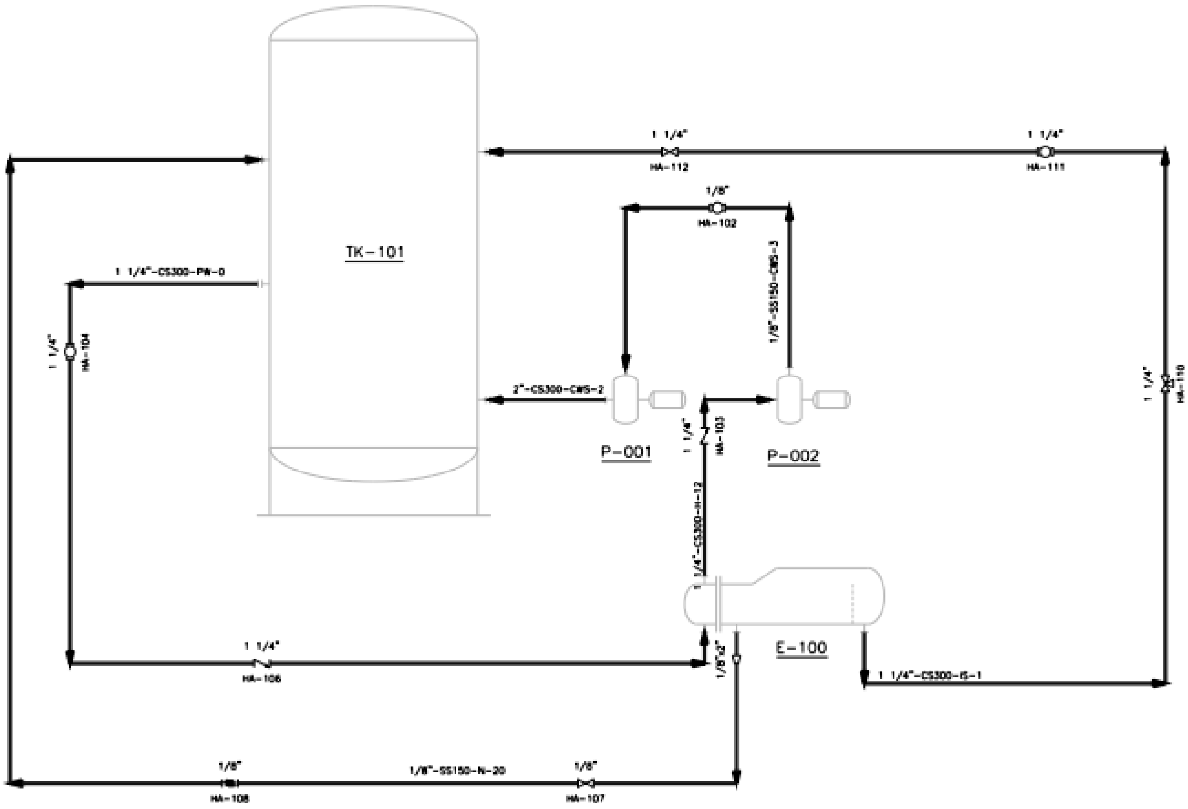
**Welcome to the ASTRO lab ML Competition!!**

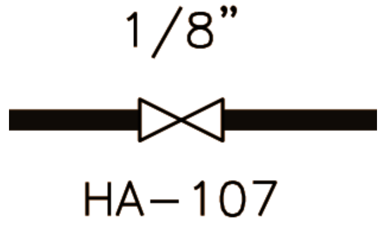
The problem you will be working on revolves around diagrams used by industrial plants known as P&IDs. These diagrams show how different pieces of equipment are connected to each other by pipes. Think about them as circuit diagrams for industrial plants. While most of the time these diagrams are created using digital software like AutoCAD, many plants will simply store an image copy. As such it is impossible to find the original files associated with the P&IDs. In order to make accurate digital twins of plants, we need to store the same information (how equipment in a plant is connected) in a more digitally consumable format (like a CSV, or other tabular format). Currently people will go through these images of P&IDs and manually convert them to a tabular format. That's a lot of work. A lot of work that has the potential to be automated. Your job is to facilitate that automation.



While we'll leave you to largely come up with the details of your implementation on your own, we've split the competition into two halves.

- Identify equipment - Approx. 2 weeks dev time
- Figure out which pipes link which pieces of equipment - Approx. 2 weeks dev time

We'll give you more details on the second part when the time comes, but for now let's focus on the first task. Equipment in a P&ID diagram for our purposes includes anything that's not a pipe. More specifically objects that are bounded in spatial dimension. These include items such as valves, tanks, pumps, etc. In the P&ID diagrams these are represented by symbols. For example a gate valve has the following symbol:



Notice how the object can be bounded entirely. Just recognizing this as a gate valve is not merely enough however. If you look at the image you’ll see some text surrounding the gate valve. We call this the tag information. The bottom line tells you the tag (i.e part ID), while the top text tells you the diameter of the pipe its fitted on. It is imperative that you focus on the relation between the text, and a given object, that is, being able to associate text in the diagram with the part itself.

The dataset contains a few P&ID drawings. These drawings may come with a legend (we will let you know which legends correspond to which drawings). These legends are a great way to understand what you’re looking at exactly. A reliance on the legend is preferred over using excessive training data. This is because the symbol set used by a particular diagram may change from diagram to diagram. Additionally the legend may give you hints as to patterns behind the text. For example strings identifying pipe information are formatted differently then strings identifying equipment information.

Given the above description of the initial problem we will judge you on the following criteria:

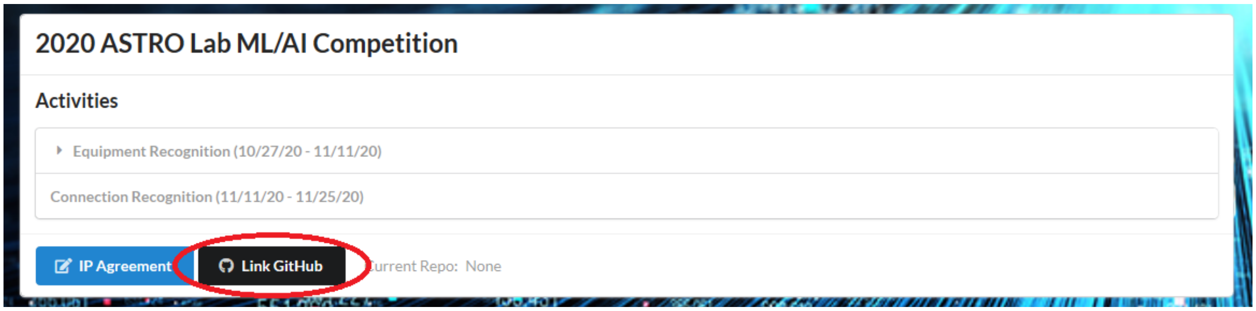
- Part Identification Accuracy
- Text Identification Accuracy
- Part and Text Association Accuracy
- Run Time (We are actually okay with longer runtimes, so long as its reasonable, waiting a few minutes to get results is fine!)

As time rolls on we will provide you with some validation data, and a test set. We’ll give you a hold out test set towards the end of the two week period, and internally we will use our own hold out test set to evaluate your work!

Some restrictions on the code you write itself:

- While this is a machine learning competition, you are free to use alternative methods such as image processing algorithms. (These are diagrams with great consistency so using algorithms is not entirely out of the picture). Just make sure that if you do choose to use an algorithm that already exists that it is open source. (For example you cannot use SIFT and SURF as those are closed source algorithms).
- Please code in Python (3.6+)

- If you’re using primarily ML code, we’re going to restrict you to using pytorch (please specify version, and clear install instructions) or Keras with Tensorflow 2.2 or TF 2.3 backend (you can also use TF 2.2 or 2.3 without keras if you want to).
- Output format
  - You are free to choose from two different output formats, the first being a CSV format, and the second being an XML format
  - The CSV format will essentially be one CSV generated per diagram. Each row will consist of the information pertaining to one detection, the first column is the part tag, or ID, the second column is the part type, the third will consist of a set of bounding box coordinates separated by a space, stored as a string, for example
    - Tag/ID,   Type,          xmin xmax ymin ymax
    - HA-107, Gate Valve, 32 102 139 209
  - The XML format follows the same output style as the LabelImg software shown here.
    - https://github.com/tzutalin/labelImg
- All your code will be uploaded to a github repository (please use your tamu enterprise account and create a repo that is publicly accessible). To specify this repository, please log in to your account at https://ml-competition.spacecraft-vr.com and go to the “Competitions” page. You can then click the “Link GitHub” button and add your repository



2020 ASTRO Lab ML/AI Competition

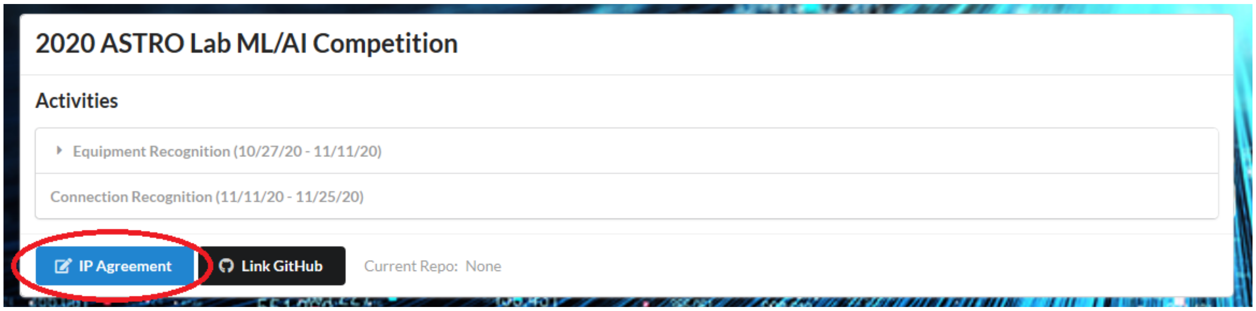
Activities

Equipment Recognition (10/27/20 - 11/11/20)

Connection Recognition (11/11/20 - 11/25/20)

Current Repo:  None

Participating in this competition implies that the ASTRO Lab has a non-exclusive right to utilize any algorithms developed by your team as part of this competition.  Please have each member of your team sign the Non-Disclosure and Intellectual Property Agreement document and email that page to Sournav@tamu.edu.  (Team members can sign their own form if that is preferred) You can download this form by logging in to your account at https://ml-competition.spacecraft-vr.com, going to the “Competitions” page, and clicking the “IP Agreement” button.



2020 ASTRO Lab ML/AI Competition

Activities

- Equipment Recognition (10/27/20 - 11/11/20)
- Connection Recognition (11/11/20 - 11/25/20)

Current Repo: None

That's all for now! If you have any questions about anything, please feel free to reach out to sournav@tamu.edu. We'll keep you updated, and we'll get you your datasets ASAP!

Happy Coding!