Plan of Attack

1. Convert CS 747 project into AMA python library.
2. Learn how to build MLOps pipeline for AMA.
3. Figure out API for MLOps pipeline.
4. Develop MLOps-as-a-Service platform.

Version 0.0.1

Description:

* Interface allows users to state what components they want their pipeline to have.
* Pipeline components are automatically set up and hosted on remote EC2.
* Users must send code and data to EC2 and manually configure their pipeline.

Pros:

* Right away, users do not have to spend time searching for open-source tools.

Cons:

* Users must still manually configure their pipeline and integrate their desired open-source tools into the pipeline.
* Users must send code and data off-prem.
* Users must make code-changes according to each individual tool. In other words, they still need to spend a lot of time and effort experimenting with each tool.

Version 0.0.2

Description:

* Interface allows users to state what components they want their pipeline to have.
* Pipeline components are automatically set up and hosted on remote EC2.
* Users must send code and data to EC2.
* Users manually mark their code for
  + Data extraction
  + Data preparation
  + Model training
  + Model evaluation
  + Model validation
  + Model deployment

using common service API.

Pros:

* Right away, users do not have to spend time searching for open-source tools.
* Users can immediately see the benefit of using MLOps platform.
* Common service API allows MLOps platform to adapt to users swapping tools.
* Users can spend less time reconfiguring pipeline and code for new tools that gets swapped in.

Cons:

* Users must still manually configure their pipeline and integrate their desired open-source tools into the pipeline.
* Users must send code and data off-prem.
* Users must make code-changes according to each individual tool.