Pio-NER

Powered by Human Ideas, delivered with AI

Publicis Sapient
Input2Output Named
Entity Recognition i.e
Pio-NER is a complete
platform for all your natural
language processing needs

While the name has Named Entity Recognition in it, it is well versed with multiple tasks such as:

1. Text2SQL for quick analysis
2. POS tagging
3. NER tagging
4. Ticketing Agent and much more!

Who will be using this tool?

Who?

- Business Users (User Interface)
- Data Scientists (Flow Environment)
- Engineering Teams (Text2sql builder for quick analysis)

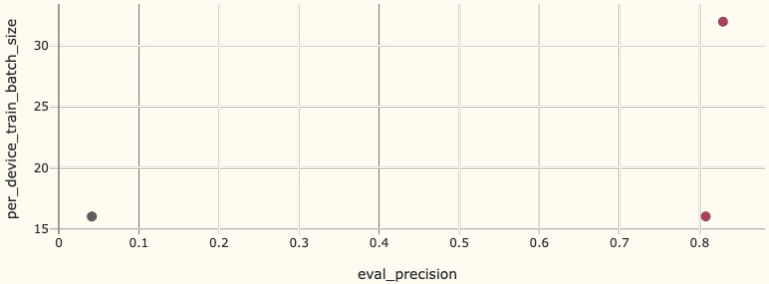
We care and love our PLANET!

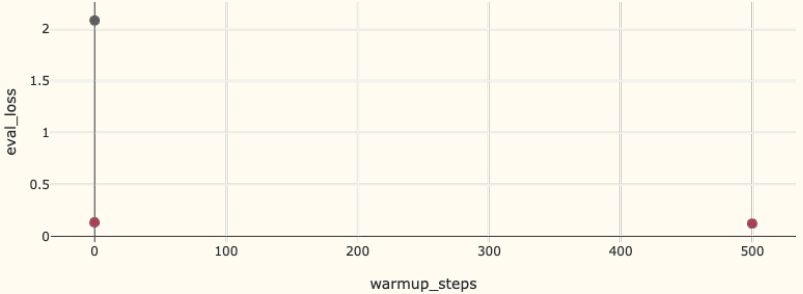
1. Algorithms are optimised using best possible methods
2. Reduce GPU footprint
3. Etc etc

Engineering Team (Internal Slides NEXT)

How we proceeded towards the solution:

1. Used Chatgpt 4o to create skeletal structure to convert the csv file into desired format, applying baseline model (dslim/bert-base-NER)
2. Integrated Notebook with MLFLOW to be able to track the training as well as be ready for invocations
3. Tracked the performance and changed the parameters for optimizing model performance. (Below are a few metrics from MLFLOW that worked and didn't work)

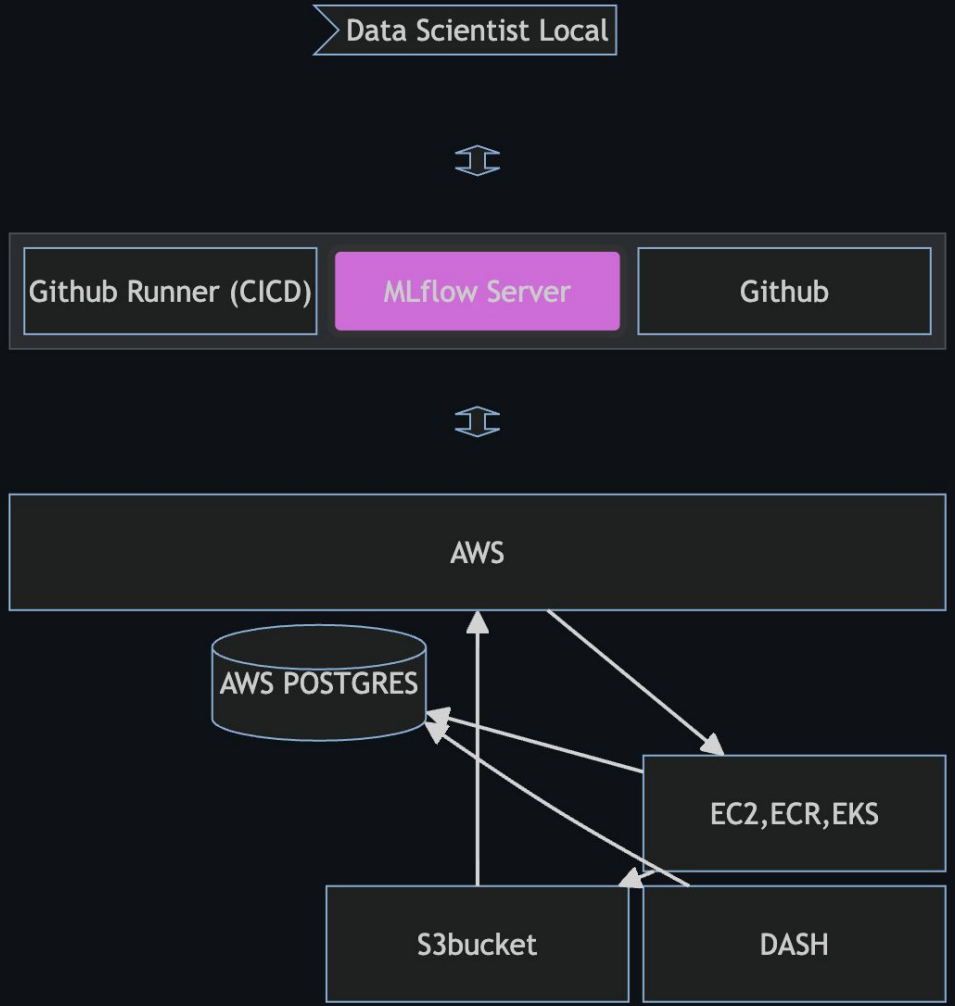




Engineering Architecture Flow Proposal

1. Data Scientists will connect to remote system through SSH with the help of Visual Studio Code and modify the training cell to tweak and make changes to model hyper parameters.
2. MLflow server running on remote will be serving the models on itself for quick tests between engineering team.
3. For Production deployment, everytime a Data Scientist retrains a model and gets its run, should be able to paste the run_id in docker file.
4. Self-hosted runner picks up this change, creates pushes the image, deploys on kubernetes.(Dockerfile gets these runs stored either on remote system/S3 and uses it to run command to serve the model with specified run.)

PROPOSED ARCHITECTURE!


Data Scientist Local
Github Runner (CICD)
MLflow Server
Github
AWS
AWS POSTGRES
EC2,ECR,EKS
S3bucket
DASH