Carnegie Mellon University

Auton Lab

An Introduction to Automated Machine Learning with Auto^{n}ML

Taking Your Machine Learning Capacity to the nth Power

Piggy Yarroll
Senior Research Programmer at Auton Lab, Carnegie Mellon University
UniForum Charter Member
Former Clout Project Postmaster
(Other fun titles)

July 25th, 2023

Outline

- What is Machine Learning?
- Why you want to use Machine Learning
- Why automate ML: Challenges of existing ML practice
- How to automate ML: **Auto^{n}ML**
- How to use **Auto^{n}ML**
- **Auto^{n}ML** Architecture
- Resources

CMU Auton Lab (www.autonlab.org) is a large Applied AI research team with 29+ years of history of developing new science and applying it to solve real-world problems.

What is Machine Learning?

- Machine learning is an umbrella term for solving problems for which development of algorithms by human programmers would be cost-prohibitive, and instead the problems are solved by helping machines 'discover' their 'own' algorithms, without needing to be explicitly told what to do by any human-developed algorithms. Wikipedia
- Machine learning includes a combination of statistical methods and heuristic algorithms to extract patterns from data.
- E.g. problem types
  - regression (curve fitting)
  - forecasting
  - classification
  - generation

What is Machine Learning?

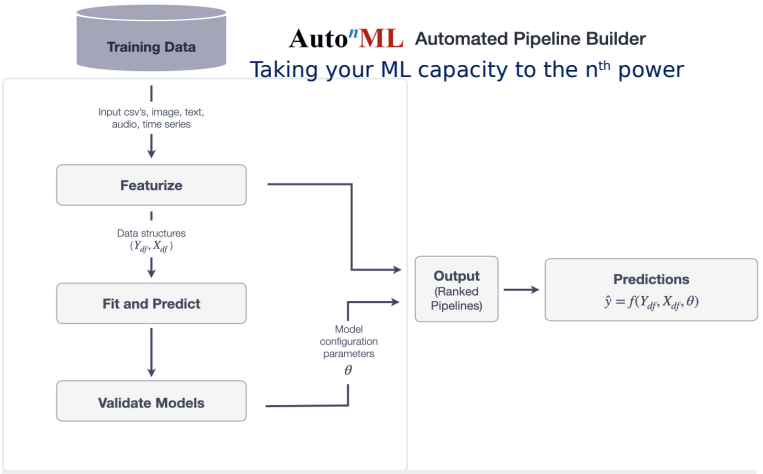
- Most algorithms provide:
  - fit() operation to learn a model from the data
  - predict() operation to predict new values
- There are a host of preprocessing and data cleaning algorithms which you may need to apply before or while you can train your model. E.g.
  - imputers
  - parsers
  - validators

What is Machine Learning? metrics

- You generally need a metric that measures the quality of your result. This metric is specific to your business case.
- Validators use the metric to rate the various results you get from your models.
- Tuners use the metric to pick the best control values (hyperparameters).

Why automate ML: Challenges of existing ML practice

- Challenges of ML in practice
  - Many opportunities but few resources in AI technology applications and deployment
- Automated machine learning (AutoML) aims to automate end-to-end process of applying machine learning to real-world data analytics problems.
- Motivations of AutoML
  - **Multiplies capacity** of Data Scientists by automating searches for plausible modeling process designs
  - **Empowers** subject-matter-experts (SMEs) such as managers and decision-makers to structure plausible problems to solve with AI
  - **Serves as a potential delivery medium** of Auton Lab ML research results



CMU Auton Lab's AutonML is an open-source software developed under DARPA D3M (Data-driven Discovery of Models) program, which supports regression, classification, forecasting, graph-based analytics of tabular, text, image, audio, and video data.

AutonML Applications

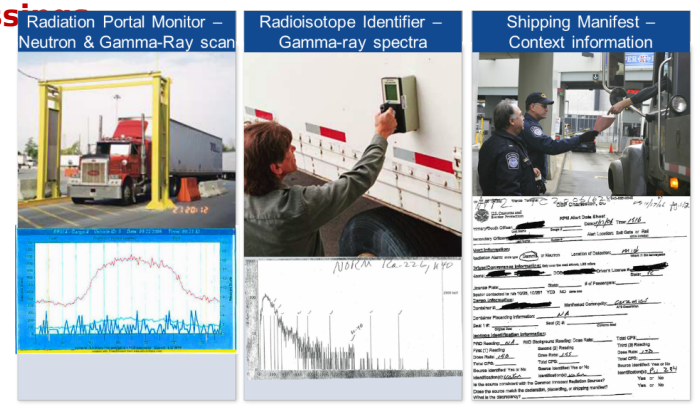
Data Scientists at the CMU Auton Lab use AutonML to:

- Identify designs with beneficial properties that can serve as alternatives to current solutions in order to validate their original design choices,
- Quickly prototype solutions to new problems and estimate level of performance that can be attained,
- Identify and qualify candidate problems for AI/ML.

All these tasks have been prohibitively labor intensive creating a risk of missing good solutions to problems at hand, and of missing good opportunities for leveraging power of AI/ML in practice.

AutonML mitigates those bottlenecks and scales up our productivity and effectiveness without the need for additional expert staff.

Example: Interdiction of radiation threat at border crossings



In the Enhanced Radiological Nuclear Inspection and Evaluation system, AI improves threat detection and significantly reduces the need for manual inspections of incoming cargo for potential radiological threat. We used AutonML to verify if the current legacy design of ERNIE AI is competitive vs. newer model type alternatives.

Example: Bedside informatics in critical care

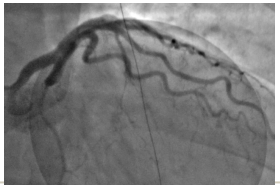
We used AutonML to assess potential utility of a few newly defined predictive tasks on high density vital sign timeseries data from intensive care, and quickly validated attainable clinical impact of AI before investing significant development efforts.

AutonML Use Cases

Example (Image Classification):
Stenosis Detection in Coronary Angiography

Goal: Predict presence and severity of stenosis in coronary angiography images

- Achieved performance on par with SOTA on a broader class of images
- Data challenges: Segmentation and localization of potential stenosis in images
  - Similar to target acquisition or anomaly detection applications of potential interest to SOCOM;
  - Closely related to the RIMFIRE task undertaken by the JAIC as part of the JAIC Joint Logistics effort;
  - Potentially extending to visual inspection of electrical wiring of Army UH-60 helicopters aimed to find irregularities such as wire chafing, loose plugs, etc.
- Human expert training time:
  - ~1 day
- AutonML model search time:
  - < 3 minutes



Example:
Predicting Engine Start Failures (UH-60@160th SOAR)

Goal: Forecast risk of over-temp on Blackhawk engine startup

- Trained using ~4TB of flight data
- Data challenges: Featurization, low prevalence of target events
- AutonML objectives: Automate featurization of data and quickly benchmark model performance of various predictive tasks

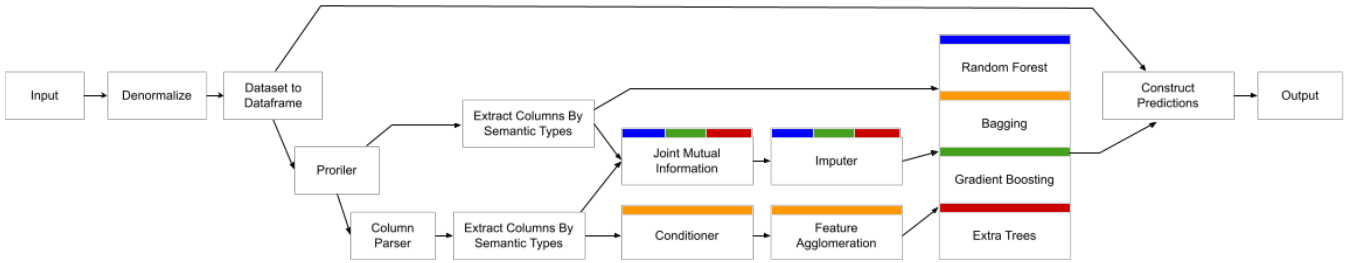
Example: Fleet of Equipment Logistics and Supply Model

Goal: Forecast part replacement and exchanges, watching exchange rate trends and giving warning of emerging part shortages

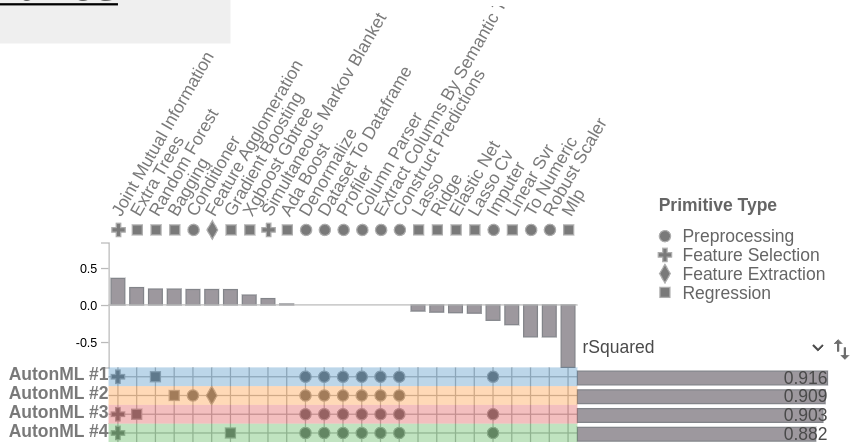
- AutonML container deployed on the Army HPC
- Modularity: The ability to produce high quality models on the fly enables widespread adoption of AI enterprise wide

Example of AutonML Pipelines on a Regression Task

Pipeline structure



Pipeline performance ranking



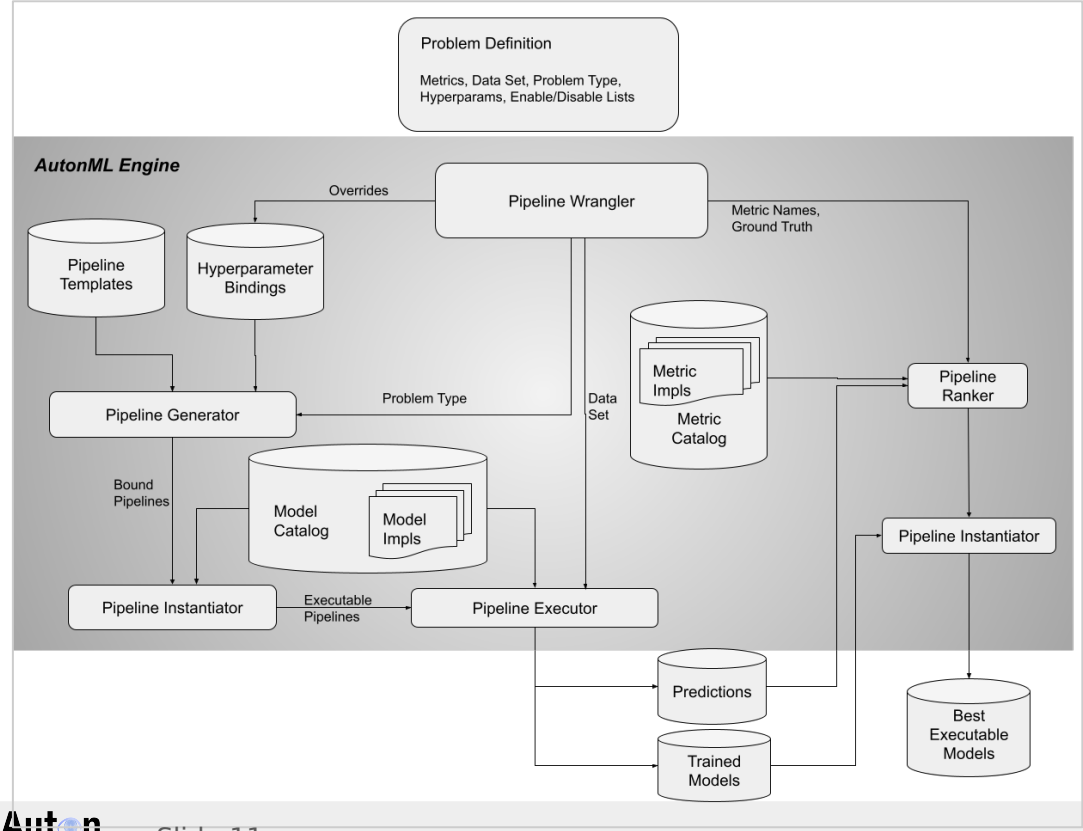
Key Definitions

- A *pipeline* is basically a series of steps that are executed in order to solve a particular problem (such as prediction based on historical data).
- A step of a pipeline is usually something that individually could, for example, transform data into another format, or fit a model for prediction.

New Design

- Continuous development of existing **Auto^{n}ML** system
  - Applying expertise acquired from DARPA D3M development
  - Designing a **new system** with higher extensibility, higher scalability and easier customization with the fast integration of:
    - Metrics catalog
    - Model algorithms catalog (e.g. MLflow)
    - Featurization catalog
    - Hyperparameter tuner catalog
    - Execution environments (e.g. cloud, AirFlow, Army HPC)
    - Problem type adapters
    - Multiple output formats (executable scripts, jupyter notebooks)
- Why we will succeed
  - Auton Lab: ML experts working closely with software architects and software engineers

New AutonML System Architecture

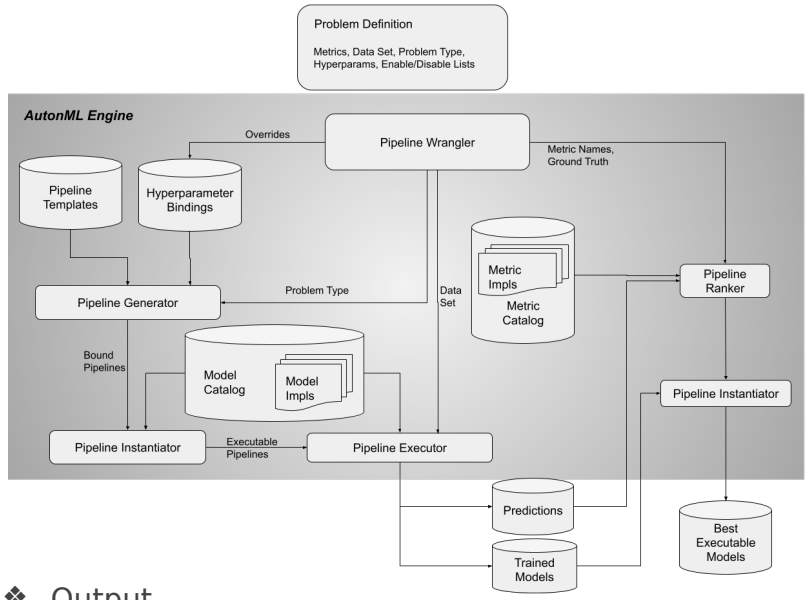


Problem definition

- Problem type: regression, classification, forecasting, etc.
- Data type: tabular, time series, image, text, audio, video, etc.
- Metrics: Mean Squared Error, Accuracy, Area Under the Curve (AUCROC), or other Key Performance Indicator (KPI) from the business perspective
- Other customizations are possible

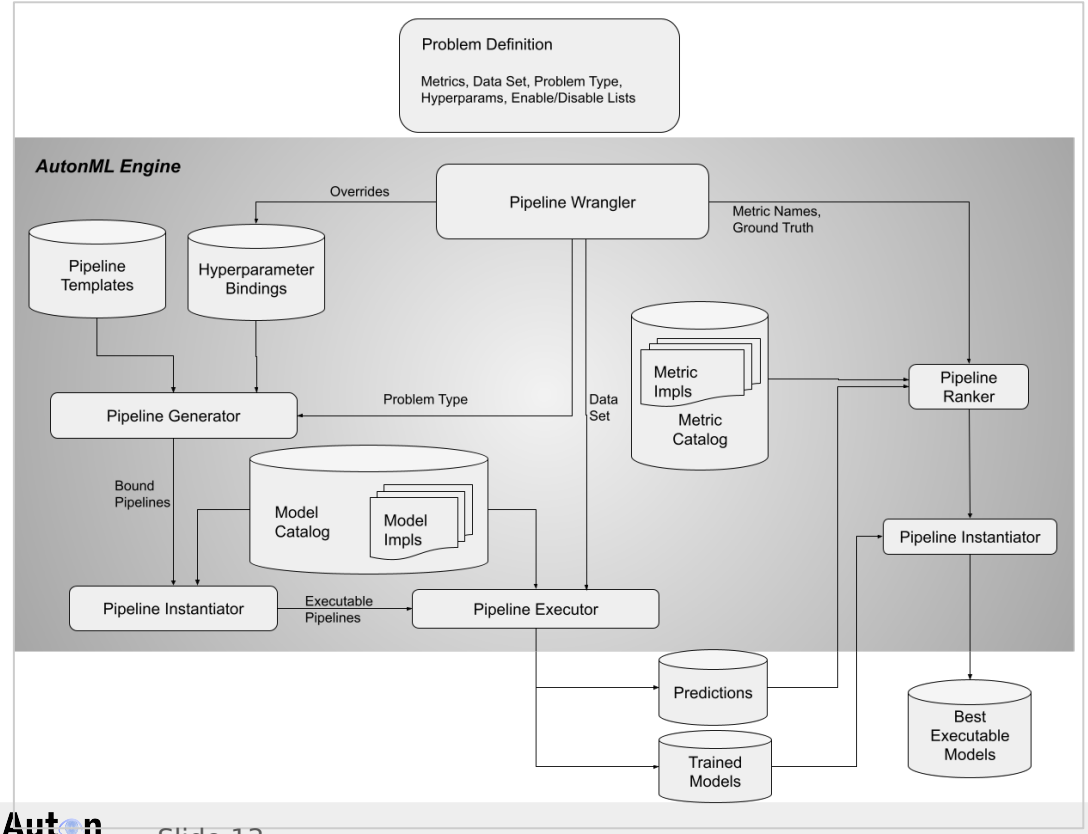
New AutonML System Architecture

- Problem definition file
  - Data type, task type, target, metric
  - Path to training and testing files
  - Customized overriding of hyperparameters, list of models to include or exclude, cross-validation settings, etc.
- Pipeline template
  - Sequence of steps that are either a specific model or a model type (e.g. preprocessing, classifier, regressor, etc.). Templates are chosen based on data type and task type.
- Model catalog
  - Model registry for primitives, each of which has one or more flexible tags indicating:
    - Usage (preprocessing, classifier, or regressor, etc.)
    - Software source (sklearn, Auton Lab, etc.)
    - Customized tags for easy grouping used by pipeline template
  - Model registration for pre-trained models
- Metric catalog
  - Metric registry for different task types
  - Customized metric registration process



- Output
  - Clearer output results
  - Training and testing predictions
  - Pipelines and rankings
  - Trained pipelines in python executables and/or .json files
  - Trained models in python object and executables (for future retrieval and prediction)
  - (Optional) Airflow-compatible objects

New AutonML System Architecture

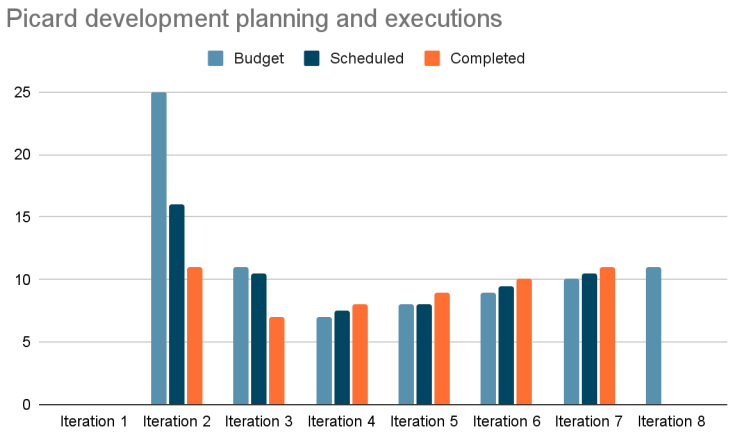


Design themes:

- Simple tasks should be simple; complicated tasks should be possible
- Catalogs for everything
  - Model catalog
  - Pipeline catalog
  - Metric catalog
  - Splitter catalog
  - Instantiator catalog
  - Executor catalog

Development Takeaways

- Software development planning
  - 8 iterations in 2 months (April 10th - June 5th, 2023)
  - 3 research programmers and 1 end user
  - Weekly planning on 9am Mondays; Mid-review on 9am Thursdays
- Weekly workload estimation and actual execution quickly align well after two iterations
  - Might need 1-2 weeks for integration and testing in the future
- Effective paired programming among 3 research programmers
- Consistent programming style, which enables short learning curve and easy debugging
  - Benefit internal and potentially external repository contributions
- Most important:
  - The development team is willing to keep working this way!



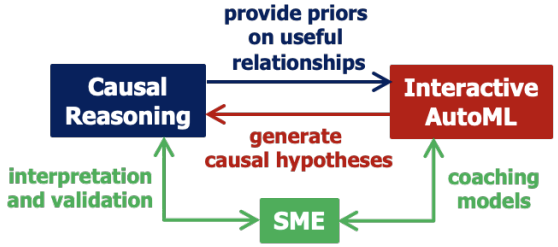
Current Status and Future Plans

Current status

- Passed v0.2 release with time series forecasting (old system fails to integrate); user starts to test the system;
- Recent development cycle focuses on flashing out more basic functionalities with more problem types, more algorithms and more featurizations
- Expected results
  - Fast integration of algorithms and featurization models from Task 1 research effort

Future

- Expected results
  - To be applied to PMx Task 2
  - Potential AI development environment (AIDE) integration
- Advanced ML functionalities
  - Incorporate model interpretation and validation, interactive weak supervision and causal discovery



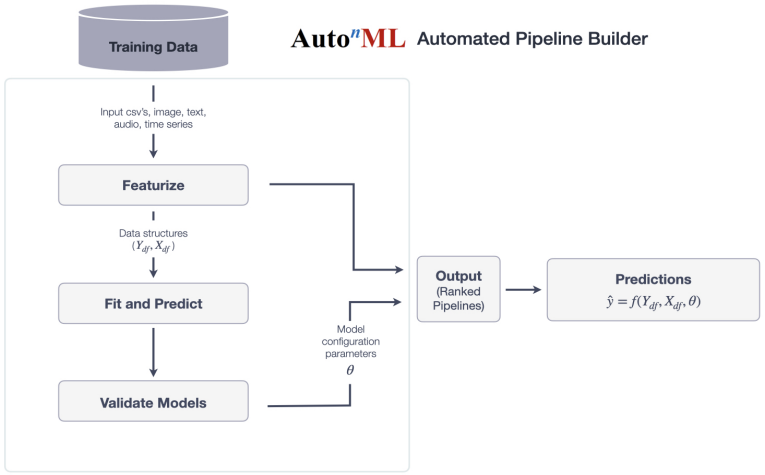
Resources

- New AutonML repository (https://gitlab.com/autonlab/ngautonml.git): coming soon
- Old AutonML Data Driven Discovery program (D3M) version
  - Gitlab repository (https://gitlab.com/autonlab/d3m/autonml.git)
  - Documentation (https://autonml.readthedocs.io/en/latest/)
- Andrew Ng's ML courses (https://www.andrewng.org/courses/)
- Other AutoML systems
  - Open-source: AutonML, H2O AutoML, TPOT (Tree-based Pipeline Optimization Tool), Auto-Sklearn, AutoGluon, etc.
  - Proprietary: DataRobot, Vertex AI (Google AutoML), Azure AutoML, etc.
- NYU's D3M Pipeline Profiler (https://github.com/VIDA-NYU/PipelineVis)

Special Thank you for Jieshi Chen for help with this slide deck and for being a great customer for the rewrite.

Auto^{n}ML

Taking your ML capacity to the nth power

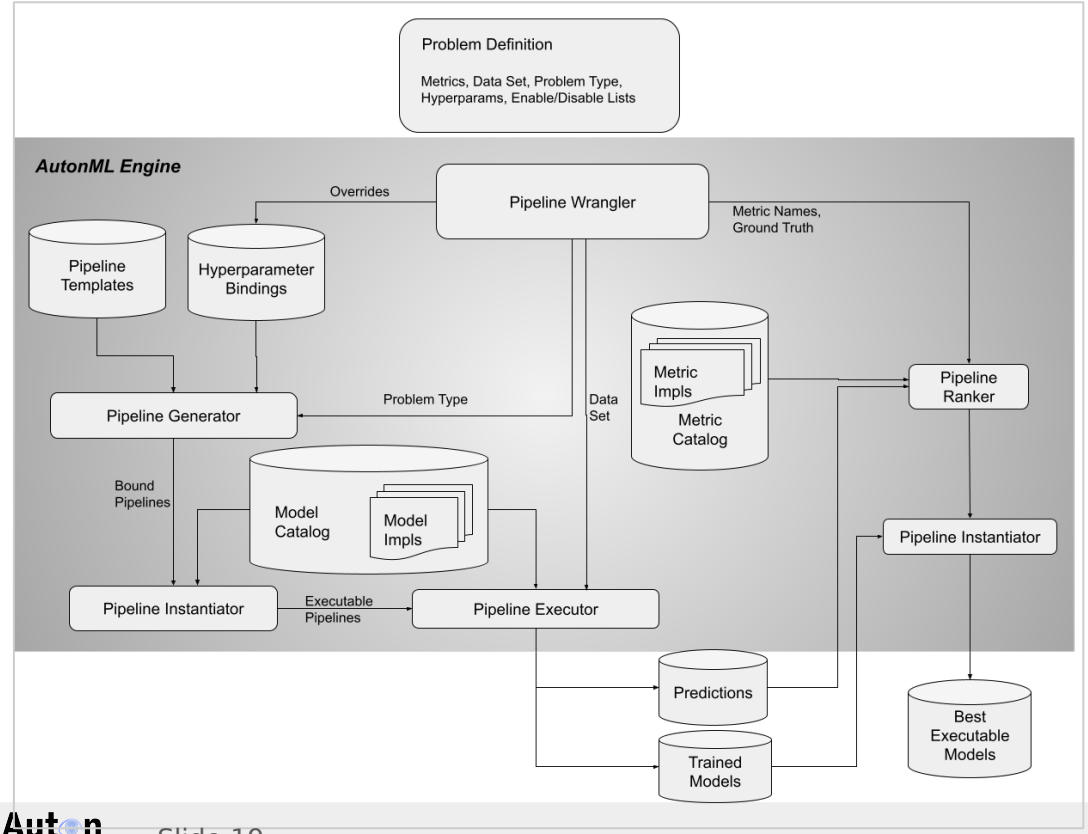


CMU Auton Lab (www.autonlab.org) is a large Applied AI research team with 29+ years of history of developing new science and applying it to solve real-world problems.

Thank you!

Appendix

New AutonML System Architecture



Wishlist:

1. Beautiful GUI
2. AutonML as a web service
3. Weak supervision
4. Interactive AutoML: LLM
5. Explainable AI