Empowering Users with AI: The Role of AutoML and Data Discovery in Data-Driven Exploration

Juliana Freire
Visualization, Imaging and Data Analysis Center (VIDA)
Computer Science & Engineering
Center for Data Science (CDS)

Joint work with: Aline Bessa, Enrico Bertini, Sonia Castelo, Gromit Yeuk-Yin Chan, Fernando Chirigati, Kyunghyun Cho, Majid Daliri, Theo Damoulas, Tamraparni Dasu, Iddo Drori, Ben Feuer, Chinmay Hegde, Yurong Liu, Roque Lopes, Chris Musco, Jorge Ono, Remi Rampin, Aecio Santos, Claudio Silva, Divesh Srivastava, Haoxiang Zhang

NYU TANDON SCHOOL OF ENGINEERING

VIDA VISUALIZATION IMAGING AND DATA ANALYSIS CENTER

AI and Machine Learning have transformed industry, academia, and government

DeepMind AI solves a half-century-old protein problem

In November it was revealed that an AI lab based in London had solved a mystery that had puzzled experts for 50 years, by predicting the 3D shape of proteins from their sequence of amino acids. Proteins, essentially the building blocks of life, are made up of amino acids.

https://www.newsweek.com/incredible-scientific-discoveries-2020-1557134

NEWS > MARKETS NEWS

Nvidia Market Cap Crosses $3 Trillion

The AI chip maker leapfrogged Apple to become the second most valuable U.S. company on

Article | Open access | Published: 07 June 2023

Health system-scale language models are all-purpose prediction engines

Lavender Yao Jiang, Xujin Chris Liu, Nima Pour Nejatian, Mustafa Nasir-Moin, Duo Wang, Anas Abidin, Kevin Eaton, Howard Antony Riina, Ilya Laufer, Paawan Punjabi, Madeline Miceli, Nora C. Kim, Cordelia Orillac, Zane Schnurman, Christopher Livia, Hannah Weiss, David Kurland, Sean Neifert, Yosef Dastagirzada, Douglas Kondziolka, Alexander T. M. Cheung, Grace Yang, Ming Cao, Mona Flores, ... Eric Karl Oermann + Show authors

Show all 28 authors for this article

Nature **619**, 357–362 (2023) | Cite this article

Sam Altman stated that the cost of training GPT-4 was more than $100 million

Analysis: How AI is helping astronomers study the universe

Science May 8, 2023 1:31 PM EDT

The famous first image of a black hole just got two times sharper. A research team used artificial intelligence to dramatically improve upon its first image from 2019, which now shows the black hole at the center of the M87 galaxy as darker and bigger than the first image depicted.

NYU TANDON SCHOOL OF ENGINEERING

Bard

Practicing Machine Learning is Hard

- Define task
- Discover/Collect Data
- Data engineering
- Modeling
- Deployment

Many tasks: classification, clustering, object detection…

Many data sources

Many ways to clean data, select features, and learn

Start from scratch for each new task

NYU TANDON SCHOOL OF ENGINEERING

VIDA VISUALIZATION IMAGING AND DATA ANALYSIS CENTER

AI in Service of Machine Learning Practice

- Define task
- Discover/Collect Data
- Data engineering
- Modeling
- Deployment

Dataset discovery

AutoML

Start from scratch for each new task

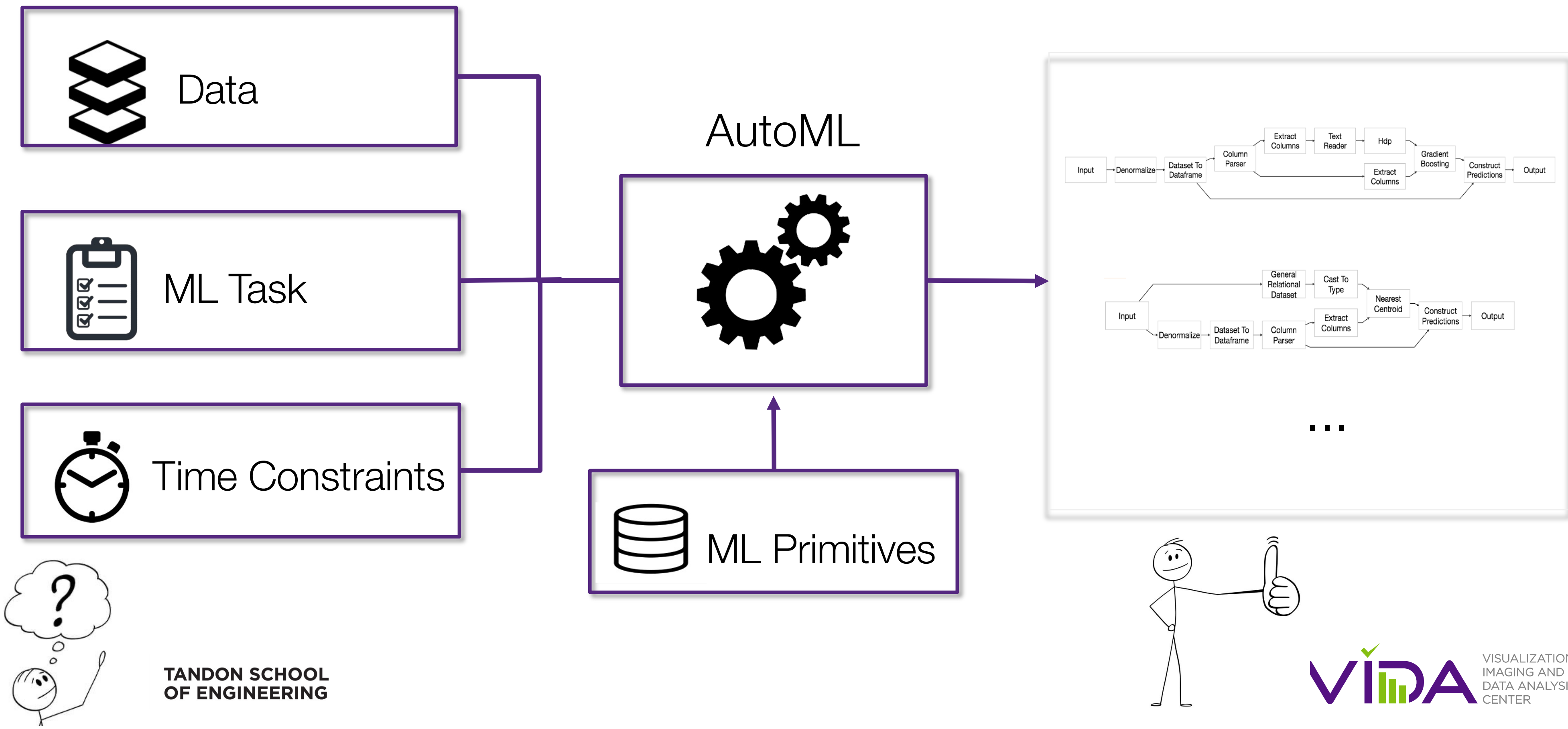
AI/ML to augment users

NYU TANDON SCHOOL OF ENGINEERING

VIDA VISUALIZATION IMAGING AND DATA ANALYSIS CENTER

AutoML: Automatic Pipeline Synthesis

AutoML outperforms humans [Hanussek et al., 2020]



TANDON SCHOOL OF ENGINEERING

VIDA VISUALIZATION IMAGING AND DATA ANALYSIS CENTER

AutoML: Automatic Pipeline Synthesis

AutoML outperforms humans [Hanussek et al., 2020]

Data

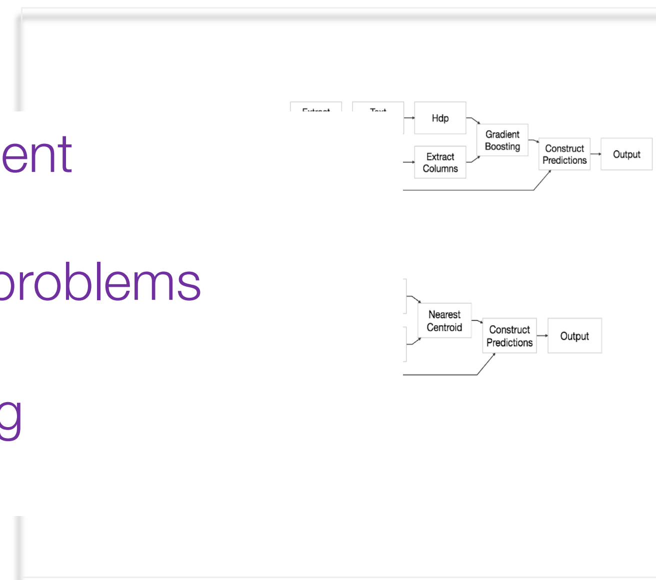
Makes data scientists more efficient

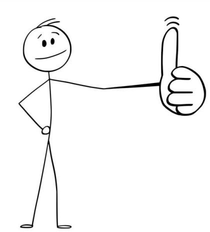
Enables engineers to work on harder problems

Democratizes machine learning

Time Constraints

ML Primitives





TANDON SCHOOL OF ENGINEERING

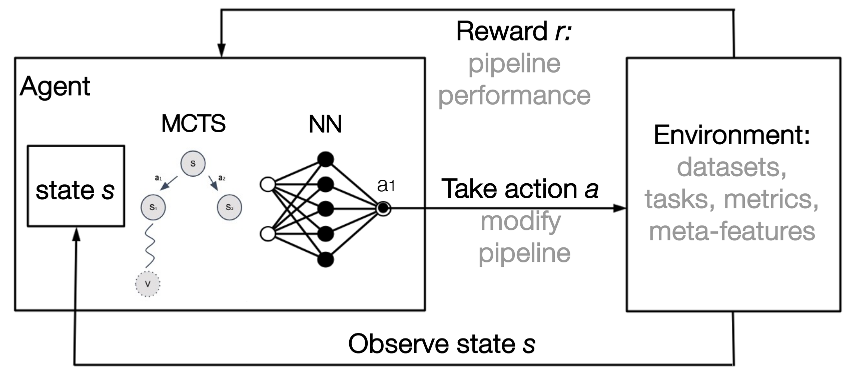
VIDA VISUALIZATION IMAGING AND DATA ANALYSIS CENTER

AutoML: Challenges

- Different problems: regression, binary classification, object detection
- Different types of data: tables, images, text
- Each (dataset, problem) combination requires different pipelines: it is expensive to construct and test a large number of pipelines -- ***too many alternatives***
  - D3M ecosystem: 312 primitives and over 1,500 hyperparameters (https://datadrivendiscovery.org)
  - Considering just the classification task over tabular data, there are 22 data cleaning, 87 data transformation, and 44 classifier primitives, leading to ***84,216 possible pipelines to test.***
- **Usability and flexibility**: enable domain experts to understand the results and customize solutions

AlphaD3M: Learning to Synthesize Pipelines

- Inspired by AlphaZero [Silver et al., 2017]:
  - Board configuration (state) → pipeline, data
  - Action → add, modify, remove primitive
- Combine Monte Carlo Tree Search (MCTS) and Neural Networks [Drori et al., AutoML 2019]
  - Given a state s the neural network predicts probabilities $P(s, a)$ over actions a from a state s
  - Produces a set of actions that describe a pipeline *p* and an estimate of its performance
  - MCTS runs and tests pipelines



Reinforcement learning

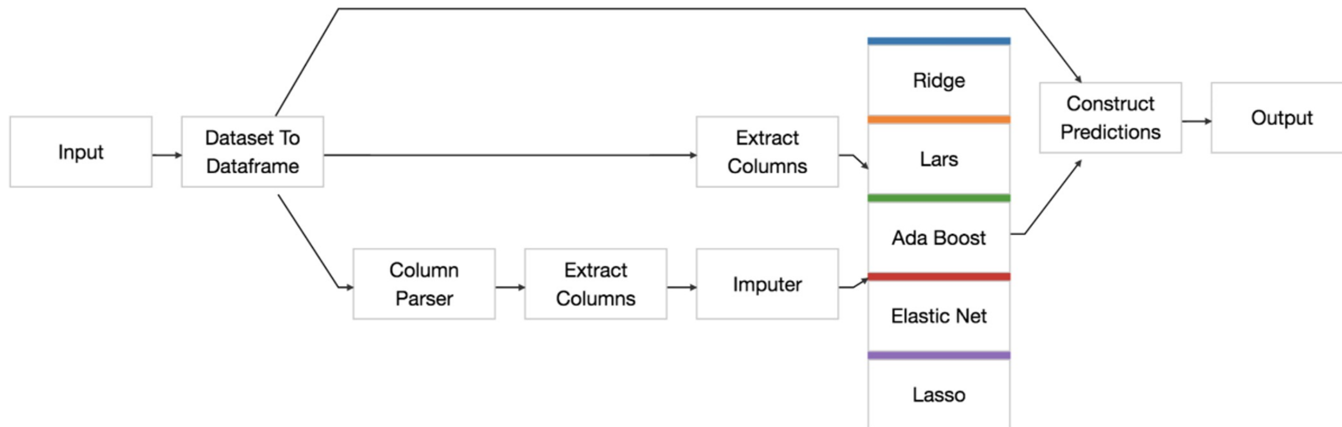
- Uses a grammar to guide the search: automatically construct the grammar through meta-learning [Lopez et al., AutoML Conf 2023]

NYU TANDON SCHOOL OF ENGINEERING

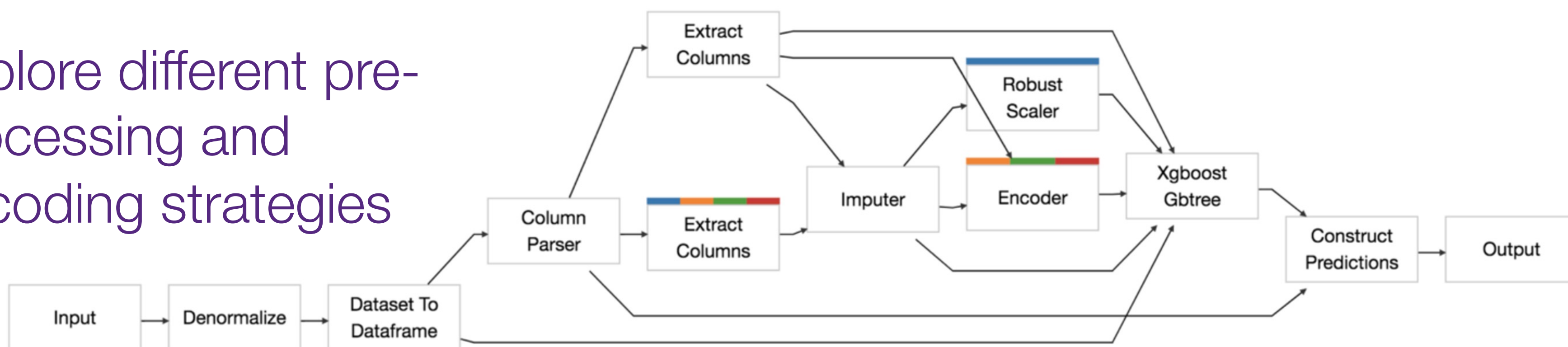
VIDA VISUALIZATION IMAGING AND DATA ANALYSIS CENTER

Exploring the Space of Pipelines: Examples

Explore different learning techniques



Explore different pre-processing and encoding strategies

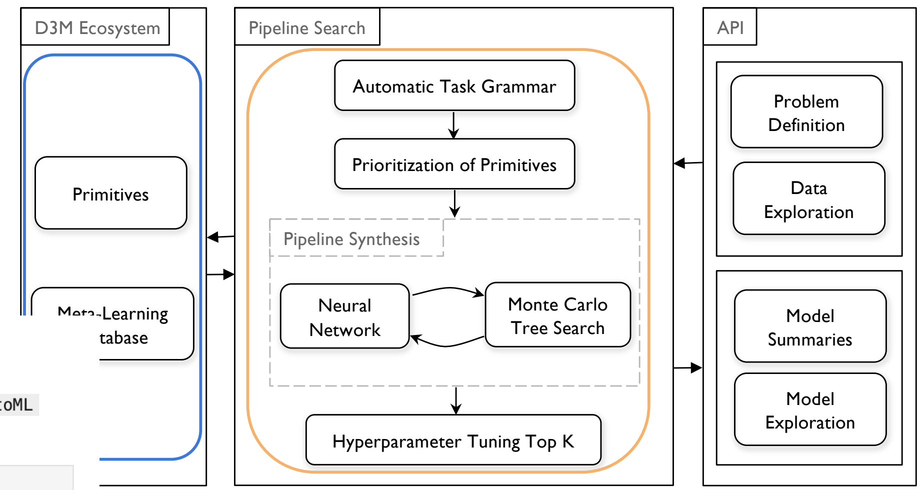


NYU TANDON SCHOOL OF ENGINEERING

VIDA VISUALIZATION IMAGING AND DATA ANALYSIS CENTER

AlphaD3M: A Multi-Task AutoML System

- Python API to build and explore ML pipelines using Jupyter notebooks
- Create models with a few lines of code



Solving Semi-supervised Classification Tasks

First, import the class `AutoML`. If you plan to use AlphaD3m via Docker/Singularity, use: `DockerAutoML` or `SingularityAutoML` classes.

In [1]:
```
from alphad3m import AutoML
# from alphad3m_containers import DockerAutoML/SingularityAutoML as AutoML
```

Generating pipelines for CSV datasets

In this example, we are generating pipelines for a CSV dataset. The SEMI_1040_sylva_prior_MIN_METADATA dataset is used for this example.

In [2]:
```
output_path = 'tmp/'
train_dataset = 'datasets/SEMI_1040_sylva_prior_MIN_METADATA/train_data.csv'
test_dataset = 'datasets/SEMI_1040_sylva_prior_MIN_METADATA/test_data.csv'
```

In [3]:
```
automl = AutoML(output_path)
automl.search_pipelines(train_dataset, time_bound=10, target='label', metric='f1', task_keywords=['semiSupervisec
```

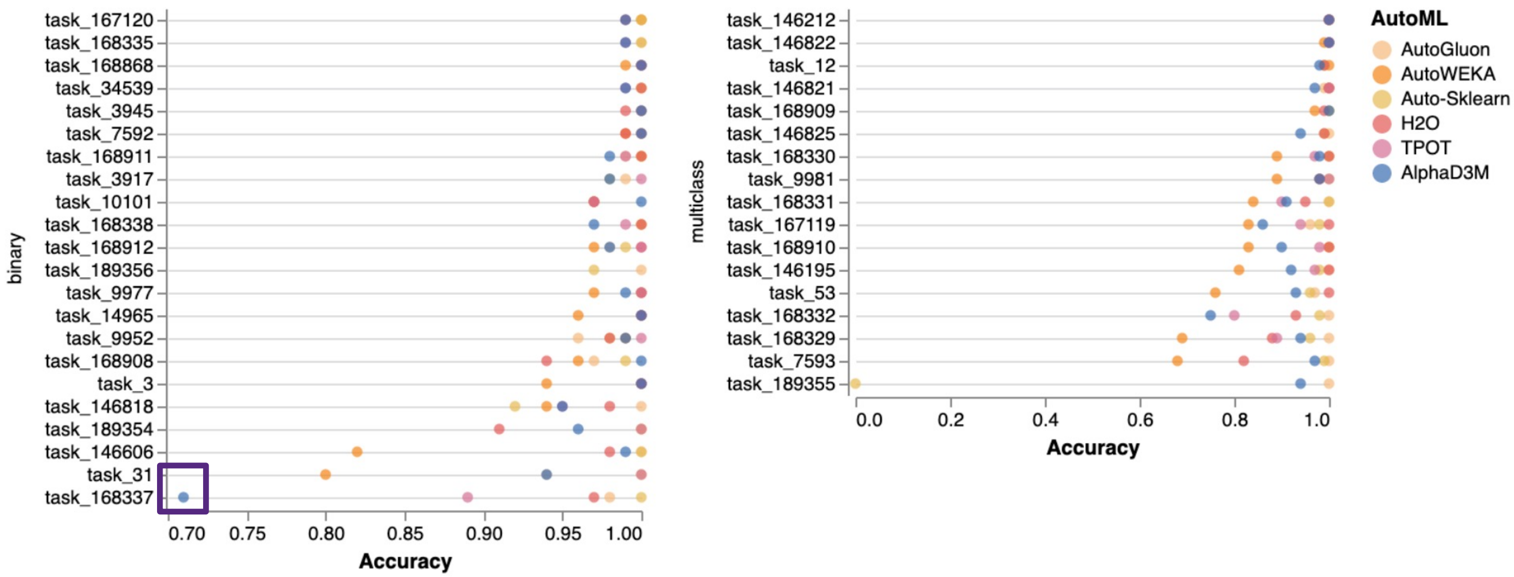
AlphaD3M: Task Coverage

- Most AutoML systems support a few tasks
- AlphaD3M supports 17 tasks and multiple data types (e.g., tabular, text, image, audio, video, graph, time series) – a benefit of its search strategy

Systems	Tabular Classification	Text classification	Image classification	Audio classification	Video classification	Tabular Regression	Clustering	Time series forecasting	Time series classification	Object detection	LUPI	Community detection	Link prediction	Graph matching	Vertex classification	Collaborative filtering	Semisupervised classification
AutoGluon	✓	✓	✓			✓				✓							
AutoWEKA	✓					✓											
Auto-Sklearn	✓					✓											
Cloud AutoML	✓	✓	✓		✓	✓				✓							
H2O	✓	✓				✓											
TPOT	✓					✓											

AlphaD3M: Performance on OpenML Benchmark

- 39 OpenML datasets represent real-world *binary and multi-class classification* problems [Gijsbers et al., 2019]
- AlphaD3M produces pipelines whose performance is on par with the other AutoML systems



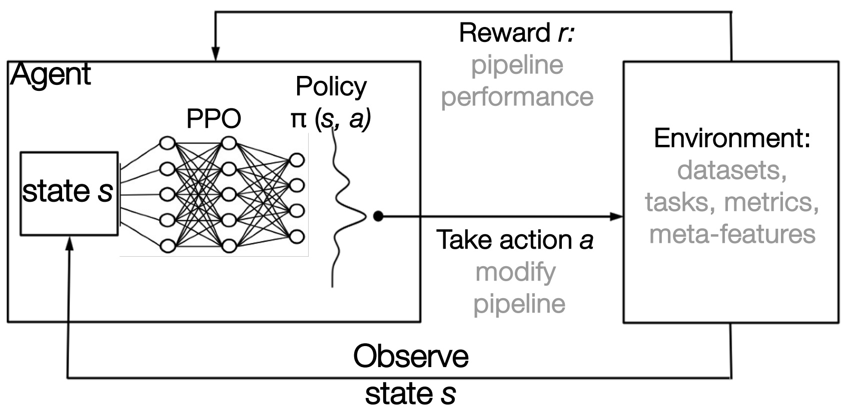
Performance of AlphaD3M could be improved with the inclusion of primitives to handle imbalanced datasets.

NYU TANDON SCHOOL OF ENGINEERING

VIDA VISUALIZATION IMAGING AND DATA ANALYSIS CENTER

From AlphaD3M to AlphaAutoML

- Explored alternative methods for performance prediction [Zhang et al., DEEM 2023] and for learning -- proximal policy optimization [Schulman et al., arxiv 2017]



AlphaAutoML learns faster and derives better pipelines than AlphaD3M

From AlphaD3M to AlphaAutoML

- Extensibility: ability to integrate new ML primitives (including LLMs) on the fly
  - Provide wrappers for HuggingFace, pytorch, fasttext, …

```
from transformers import AutoTokenizer, AutoModel
```

```
from alpha_automl.wrapper_primitives.huggingface import HuggingfaceInterface
```

```
automl = AutoMLClassifier(output_path, time_bound=40, verbose=True)
```

```
tokenizer = AutoTokenizer.from_pretrained("cardiffnlp/twitter-roberta-base-sentiment")
model = AutoModel.from_pretrained("cardiffnlp/twitter-roberta-base-sentiment", output_hidden_states=True)
model_name = 'cardiffnlp/twitter-roberta-base-sentiment'
my_tweet_embedder = HuggingfaceInterface(model, tokenizer, 'cardiffnlp/twitter-roberta-base-sentiment', last_four
```

```
automl.add_primitives([(my_tweet_embedder, 'TEXT_ENCODER')])
```

From AlphaD3M to AlphaAutoML

- More efficient learning method and extensibility – ability to keep up with new ML methods
- Usability and interoperability: Compatible with standard Python ML libraries, e.g., sklearn, pytorch, etc.
- Pip installable

```
pip install alpha-automl
```

```
In [1]: from alpha_automl import AutoMLTimeSeries

In [2]: automl = AutoMLTimeSeries(output_path, time_bound=10,
                                  date_column="Date", target_column="Close")
        automl.fit(X_train, y_train)
```

```
In [1]: from alpha_automl import AutoMLClassifier

In [2]: automl = AutoMLClassifier(output_path, time_bound=10)
        automl.fit(X_train, y_train)
```

NYU TANDON SCHOOL OF ENGINEERING

https://github.com/VIDA-NYU/alpha-automl

VIDA VISUALIZATION IMAGING AND DATA ANALYSIS CENTER

AlphaAutoML Performance: Preliminary Results

In [5]:
```
ranks = calculate_rank(performances)
ranks.sort_values(by='average_rank')
```

Out[5]:

	average_rank	task_10101	task_12	task_146195	task_146212	task_146606	task_146818	task_146821	task_146822	task_146825	...
Auto-Sklearn	2.79	3.0	6.0	3.0	1.0	8.0	3.0	1.0	3.0	2.0	...
Alpha-AutoML	3.49	3.0	2.0	4.0	1.0	6.0	3.0	1.0	1.0	3.0	...
AutoGluon	3.51	3.0	6.0	2.0	7.0	1.0	1.0	7.0	3.0	1.0	...
H2O	3.69	3.0	5.0	1.0	1.0	5.0	2.0	1.0	3.0	4.0	...
TPOT	4.08	2.0	1.0	5.0	1.0	4.0	3.0	1.0	3.0	8.0	...
AlphaD3M	5.10	1.0	8.0	7.0	1.0	3.0	6.0	8.0	3.0	5.0	...
Alpha-AutoML_Old	5.15	8.0	2.0	6.0	1.0	2.0	7.0	1.0	8.0	7.0	...
AutoWEKA	6.56	3.0	2.0	8.0	8.0	7.0	8.0	6.0	1.0	6.0	...

8 rows × 40 columns

Human-Centered AutoML

- Automation is not enough
- How to evaluate and compare pipelines?
  - Efficiency
  - Correctness

- How to improve pipelines?
  - Customize the pipelines
  - Improve the data (Data-Centric AI – https://datacentricai.org)

- Support domain experts

In [4]:
```
automl.plot_leaderboard()
```

Out[4]:

ranking	id	summary	f1_macro
1	28703d90-da87-4662-a159-0f0223340e5b	add_semantic_types.common, imputer.sklearn, corex_text.dsbox, one_hot_encoder.sklearn, to_numeric.dsbox, iqr_scaler.dsbox, select_percentile.sklearn, logistic_regression.sklearn	0.623430
2	bd0f1730-72e9-4f56-81c7-521a4446dff9	add_semantic_types.common, imputer.sklearn, encoder.distiltextencoder, one_hot_encoder.sklearn, to_numeric.dsbox, max_abs_scaler.sklearn, variance_threshold.sklearn, xgboost_gbtree.common	0.604560
3	f57d6e5e-ed32-4a23-9390-9fa7f41fcfa8	add_semantic_types.common, imputer.sklearn, encoder.distiltextencoder, one_hot_encoder.sklearn, to_numeric.dsbox, iqr_scaler.dsbox, select_percentile.sklearn, xgboost_dart.common	0.589560
4	6ade3bf3-9bbe-4256-bdc5-3d8a60041bca	add_semantic_types.common, imputer.sklearn, encoder.distiltextencoder, one_hot_encoder.sklearn, to_numeric.dsbox, max_abs_scaler.sklearn, select_percentile.sklearn, xgboost_dart.common	0.589560
5	fb4c91a0-bcb8-4ef2-9edd-3236167ee03e	add_semantic_types.common, imputer.sklearn, tfidf_vectorizer.sklearn, one_hot_encoder.sklearn, to_numeric.dsbox, iqr_scaler.dsbox, select_percentile.sklearn, xgboost_dart.common	0.576570
6	381ba1ea-be88-4418-9a30-288525201e28	add_semantic_types.common, imputer.sklearn, tfidf_vectorizer.sklearn, one_hot_encoder.sklearn, to_numeric.dsbox, max_abs_scaler.sklearn, select_percentile.sklearn, xgboost_dart.common	0.576570
7	94a31aa6-9569-4ae6-bd3b-fc52b59cfccc	add_semantic_types.common, imputer.sklearn, tfidf_vectorizer.sklearn, encoder.dsbox, to_numeric.dsbox, max_abs_scaler.sklearn, select_percentile.sklearn, xgboost_dart.common	0.576570
8	2a4656f8-d5d0-48a3-b7f1-a8a1540c4101	add_semantic_types.common, imputer.sklearn, corex_text.dsbox, one_hot_encoder.sklearn, to_numeric.dsbox, iqr_scaler.dsbox, select_percentile.sklearn, xgboost_dart.common	0.569470
9	467cace3-e895-4850-8773-706b3dc4891e	add_semantic_types.common, imputer.sklearn, corex_text.dsbox, one_hot_encoder.sklearn, to_numeric.dsbox, max_abs_scaler.sklearn, select_percentile.sklearn, xgboost_dart.common	0.569470
10	702e3524-0aaf-4b83-801a-f47141b699e9	add_semantic_types.common, imputer.sklearn, corex_text.dsbox, encoder.dsbox, to_numeric.dsbox, max_abs_scaler.sklearn, select_percentile.sklearn, xgboost_dart.common	0.569470
11	d90d24f2-39b3-46db-ba53-9f3601fa9fc0	add_semantic_types.common, imputer.sklearn, tfidf_vectorizer.sklearn, encoder.dsbox, xgboost_gbtree.common	0.566240
12	5a892918-a90e-4552-b1fe-2fecd03c9f07	add_semantic_types.common, imputer.sklearn, tfidf_vectorizer.sklearn, one_hot_encoder.sklearn, to_numeric.dsbox, max_abs_scaler.sklearn, variance_threshold.sklearn, xgboost_gbtree.common	0.566240
13	8d0e8756-0294-41a2-b7b1-c718b56d0cf4	add_semantic_types.common, imputer.sklearn, tfidf_vectorizer.sklearn, encoder.dsbox, to_numeric.dsbox, max_abs_scaler.sklearn, variance_threshold.sklearn, xgboost_gbtree.common	0.566240
14	990e35e6-44b1-4516-aa43-2dbfbd5544b4	add_semantic_types.common, imputer.sklearn, tfidf_vectorizer.sklearn, encoder.dsbox, gradient_boosting.sklearn	0.563780

Human-Centered AutoML

- Automation is not enoug
- How to evaluate and cor
    - Efficiency
    - Correctness
    - Agreement with human judge
- How to improve pipeline:
    - Customize the pipelines
    - Improve the data

        Data-Centric AI--https://da
- Support domain experts

In [4]:
```
automl.plot_leaderboard()
```

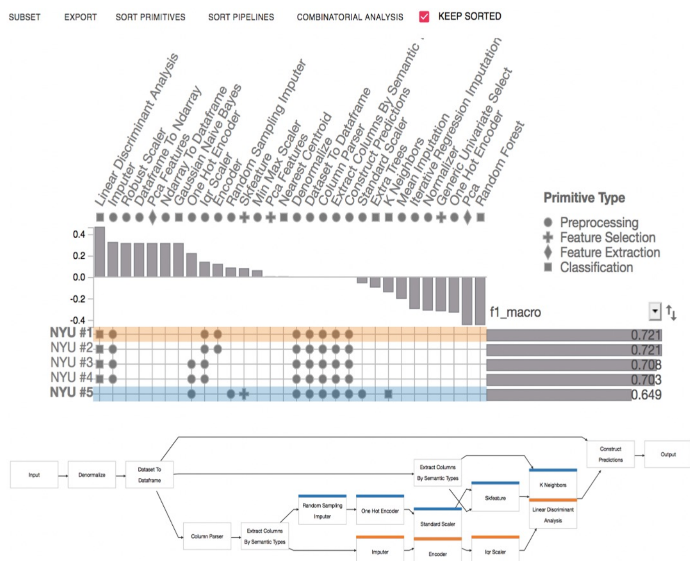
Out[4]:

ranking	id	summary	f1_macro
1	28703d90-da87-4662-a159-0f0223340e5b	add_semantic_types.common, imputer.sklearn, corex_text.dsbox, one_hot_encoder.sklearn, to_numeric.dsbox, iqr_scaler.dsbox, select_percentile.sklearn, logistic_regression.sklearn	0.623430
2	bd0f1730-72e9-4f56-81c7-521a4446dff9	add_semantic_types.common, imputer.sklearn, encoder.distiltextencoder, one_hot_encoder.sklearn, to_numeric.dsbox, max_abs_scaler.sklearn, variance_threshold.sklearn, xgboost_gbtree.common	0.604560
3	f57d6e5e-ed32-4a23-9390-9fa7f41fcfa8	add_semantic_types.common, imputer.sklearn, encoder.distiltextencoder, one_hot_encoder.sklearn, to_numeric.dsbox, iqr_scaler.dsbox, select_percentile.sklearn, xgboost_dart.common	0.589560
4	6ade3bf3-9bbe-4256-bdc5-3d8a60041bca	add_semantic_types.common, imputer.sklearn, encoder.distiltextencoder, one_hot_encoder.sklearn, to_numeric.dsbox, max_abs_scaler.sklearn, select_percentile.sklearn, xgboost_dart.common	0.589560
5	fb4c91a0-bcb8-4ef2-9edd-3236167ee03e	add_semantic_types.common, imputer.sklearn, tfidf_vectorizer.sklearn, one_hot_encoder.sklearn, to_numeric.dsbox, iqr_scaler.dsbox, select_percentile.sklearn, xgboost_dart.common	0.576570
6	381ba1ea-be88-4418-9a30-288525201e28	add_semantic_types.common, imputer.sklearn, tfidf_vectorizer.sklearn, one_hot_encoder.sklearn, to_numeric.dsbox, max_abs_scaler.sklearn, select_percentile.sklearn, xgboost_dart.common	0.576570
7	94a31aa6-9569-4ae6-bd3b-fc52b59cfccc	add_semantic_types.common, imputer.sklearn, tfidf_vectorizer.sklearn, encoder.dsbox, to_numeric.dsbox, max_abs_scaler.sklearn, select_percentile.sklearn, xgboost_dart.common	0.576570
8	2a4656f8-d5d0-48a3-b7f1-a8a1540c4101	add_semantic_types.common, imputer.sklearn, corex_text.dsbox, one_hot_encoder.sklearn, to_numeric.dsbox, iqr_scaler.dsbox, select_percentile.sklearn, xgboost_dart.common	0.569470
9	467cace3-e895-4850-8773-706b3dc4891e	add_semantic_types.common, imputer.sklearn, corex_text.dsbox, one_hot_encoder.sklearn, to_numeric.dsbox, max_abs_scaler.sklearn, select_percentile.sklearn, xgboost_dart.common	0.569470
10	702e3524-0aaf-4b83-801a-f47141b699e9	add_semantic_types.common, imputer.sklearn, corex_text.dsbox, encoder.dsbox, to_numeric.dsbox, max_abs_scaler.sklearn, select_percentile.sklearn, xgboost_dart.common	0.569470
11	d90d24f2-39b3-46db-ba53-9f3601fa9fc0	add_semantic_types.common, imputer.sklearn, tfidf_vectorizer.sklearn, encoder.dsbox, xgboost_gbtree.common	0.566240
12	5a892918-a90e-4552-b1fe-2fecd03c9f07	add_semantic_types.common, imputer.sklearn, tfidf_vectorizer.sklearn, one_hot_encoder.sklearn, to_numeric.dsbox, max_abs_scaler.sklearn, variance_threshold.sklearn, xgboost_gbtree.common	0.566240
13	8d0e8756-0294-41a2-b7b1-c718b56d0cf4	add_semantic_types.common, imputer.sklearn, tfidf_vectorizer.sklearn, encoder.dsbox, to_numeric.dsbox, max_abs_scaler.sklearn, variance_threshold.sklearn, xgboost_gbtree.common	0.566240
14	990e35e6-44b1-4516-aa43-2dbfbd5544b4	add_semantic_types.common, imputer.sklearn, tfidf_vectorizer.sklearn, encoder.dsbox, gradient_boosting.sklearn	0.563780

summary	f1_macro
ypes.common, imputer.sklearn, corex_text.dsbox, lsbox, iqr_scaler.dsbox, select_percentile.sklearn, logistic_regression.sklearn	0.623430
mmon, imputer.sklearn, encoder.distiltextencoder, learn, to_numeric.dsbox, max_abs_scaler.sklearn, ance_threshold.sklearn, xgboost_gbtree.common	0.604560
mmon, imputer.sklearn, encoder.distiltextencoder, lsbox, iqr_scaler.dsbox, select_percentile.sklearn, xgboost_dart.common	0.589560
mmon, imputer.sklearn, encoder.distiltextencoder, learn, to_numeric.dsbox, max_abs_scaler.sklearn, select_percentile.sklearn, xgboost_dart.common	0.589560
common, imputer.sklearn, tfidf_vectorizer.sklearn, lsbox, iqr_scaler.dsbox, select_percentile.sklearn, xgboost_dart.common	0.576570
common, imputer.sklearn, tfidf_vectorizer.sklearn, learn, to_numeric.dsbox, max_abs_scaler.sklearn, select_percentile.sklearn, xgboost_dart.common	0.576570
r.sklearn, tfidf_vectorizer.sklearn, encoder.dsbox, nax_abs_scaler.sklearn, select_percentile.sklearn, xgboost_dart.common	0.576570
ypes.common, imputer.sklearn, corex_text.dsbox, lsbox, iqr_scaler.dsbox, select_percentile.sklearn, xgboost_dart.common	0.569470
ypes.common, imputer.sklearn, corex_text.dsbox, learn, to_numeric.dsbox, max_abs_scaler.sklearn, select_percentile.sklearn, xgboost_dart.common	0.569470
mputer.sklearn, corex_text.dsbox, encoder.dsbox, nax_abs_scaler.sklearn, select_percentile.sklearn, xgboost_dart.common	0.569470
r.sklearn, tfidf_vectorizer.sklearn, encoder.dsbox, xgboost_gbtree.common	0.566240
common, imputer.sklearn, tfidf_vectorizer.sklearn, learn, to_numeric.dsbox, max_abs_scaler.sklearn, ance_threshold.sklearn, xgboost_gbtree.common	0.566240
r.sklearn, tfidf_vectorizer.sklearn, encoder.dsbox, x_abs_scaler.sklearn, variance_threshold.sklearn, xgboost_gbtree.common	0.566240
r.sklearn, tfidf_vectorizer.sklearn, encoder.dsbox, gradient_boosting.sklearn	0.563780

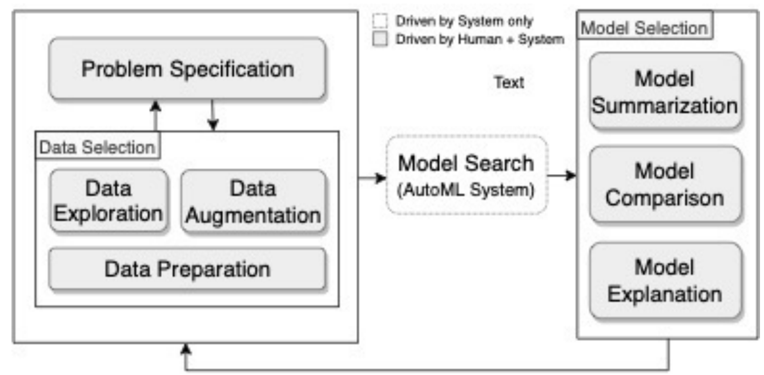
Usability, Explainability, and Trust

Pipeline Profiler: support data scientists -- explore pipelines



[Ono et al., IEEE Vis 2019]
https://github.com/VIDA-NYU/PipelineVis

Visus: support domain experts practice machine learning



[Santos et al., HILDA 2019]
https://github.com/VIDA-NYU/PipelineVis

NYU TANDON SCHOOL OF ENGINEERING

VIDA VISUALIZATION IMAGING AND DATA ANALYSIS CENTER

Pipeline Profiler: Exploring AutoML Pipelines

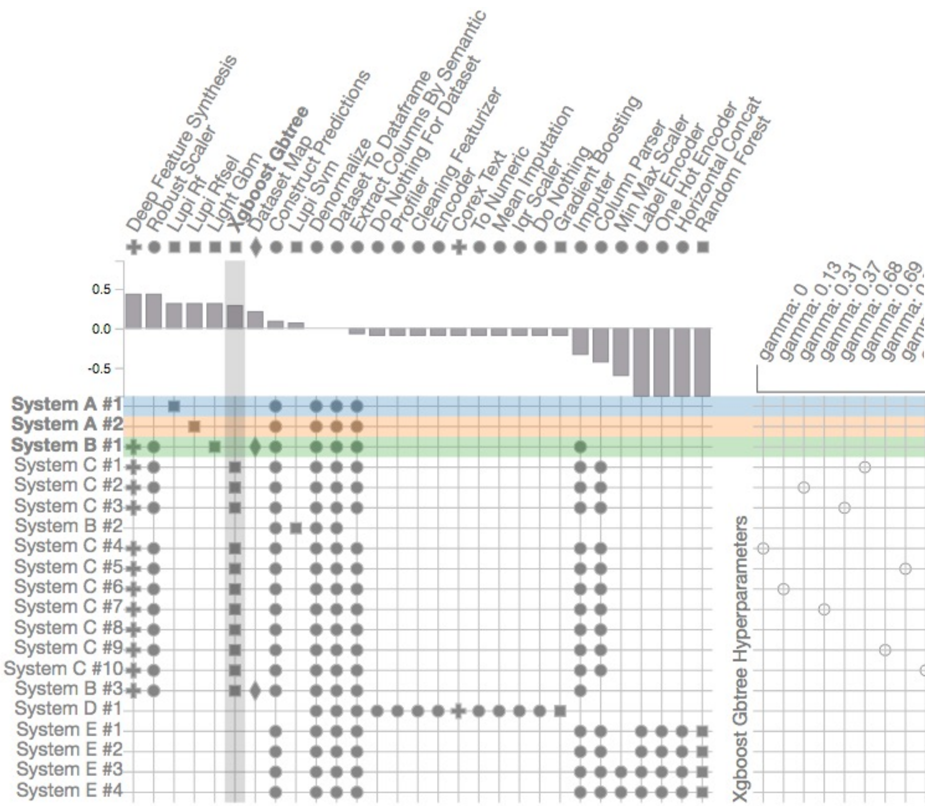
- AutoML systems lack transparency: How to explore and build trust results they produce? [Xin et al., ACM CHI 2021]
- DARPA D3M program: 20+ research groups working on AutoML
  - Systems shared the same infrastructure: API, 300 primitives (Python), pipeline description language (DAG – JSON) https://www.darpa.mil/program/data-driven-discovery-of-models

Hard to compare and understand pipelines produced for a problem

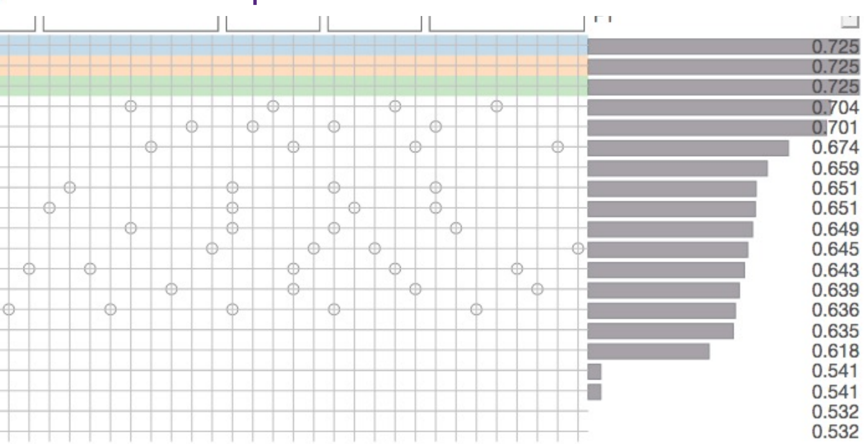
- *User interviews:* Routinely explored pipeline collections; reading text files one at a time is tedious; DAG pipeline structure is hard to grasp

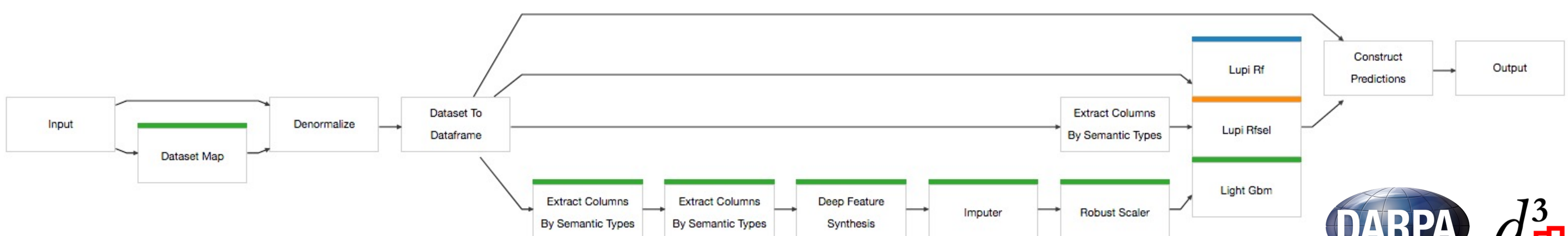
- PipelineProfiler: A visualization library designed together with D3M experts to enable the exploration and comparison of pipelines derived by AutoML systems

[Ono et al., IEEE Vis 2019]
https://github.com/VIDA-NYU/PipelineVis

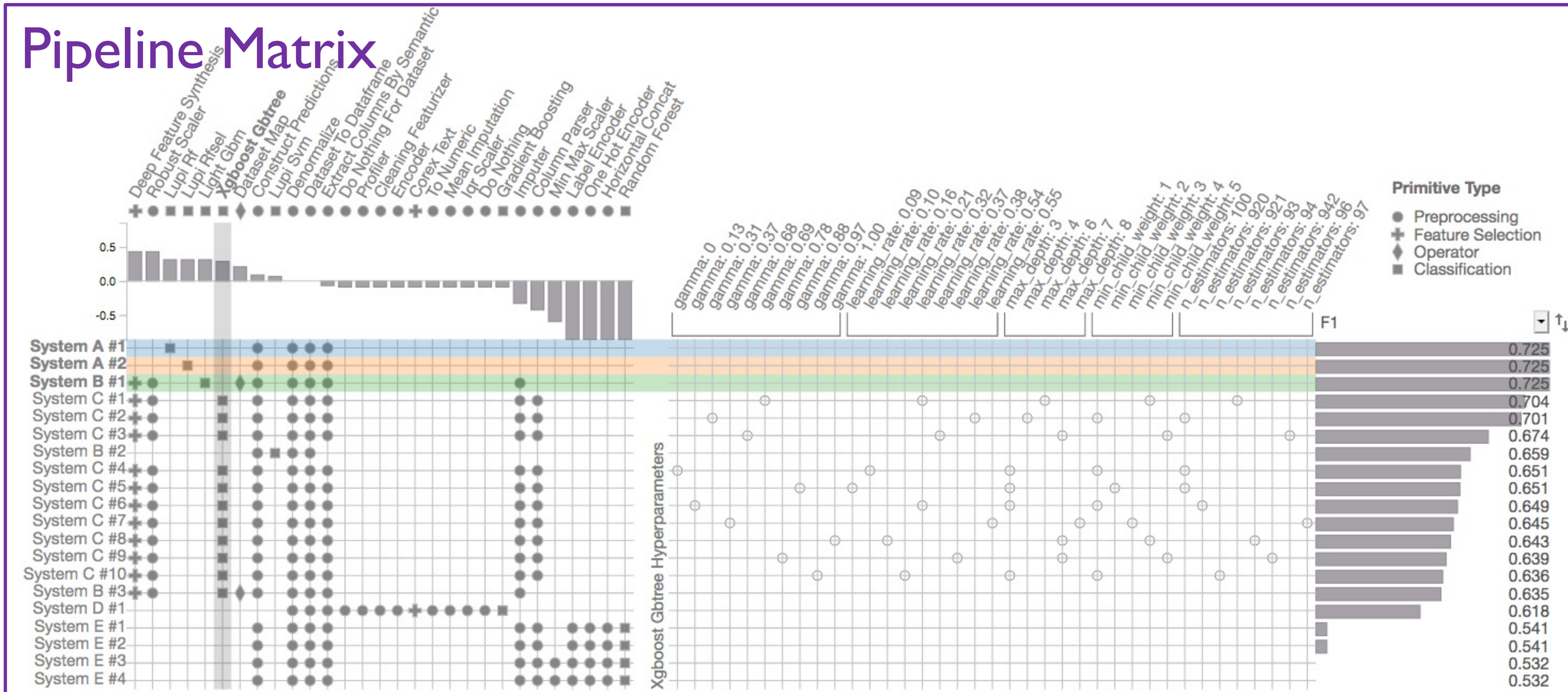


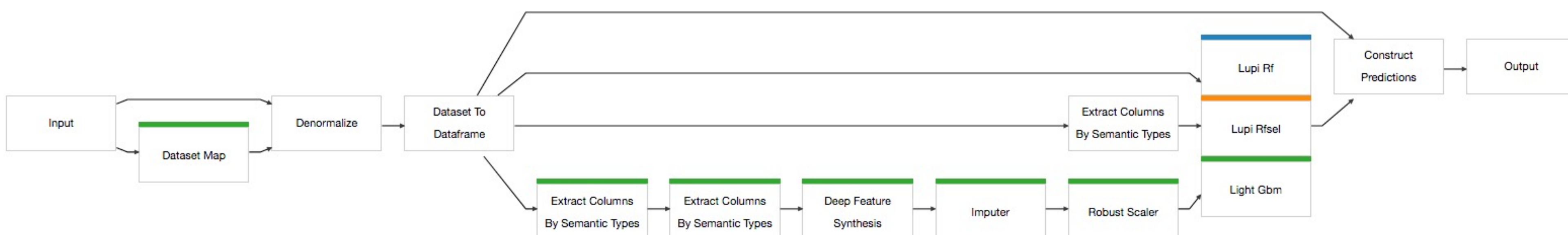
PipelineProfiler runs inside Jupyter Notebooks and provides an interactive visual analytics interface to help data scientists gain insights into pipelines and how to improve them



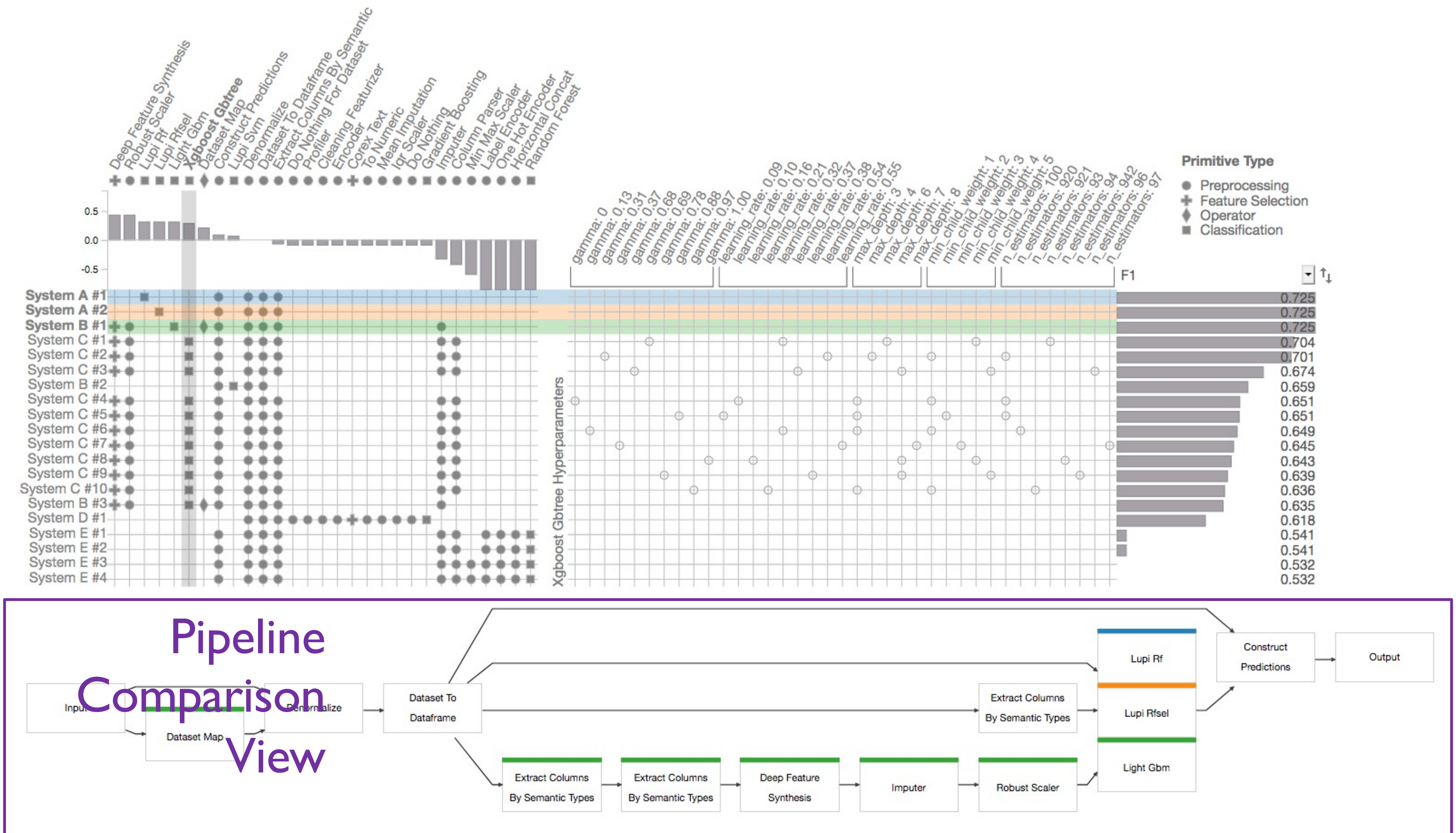


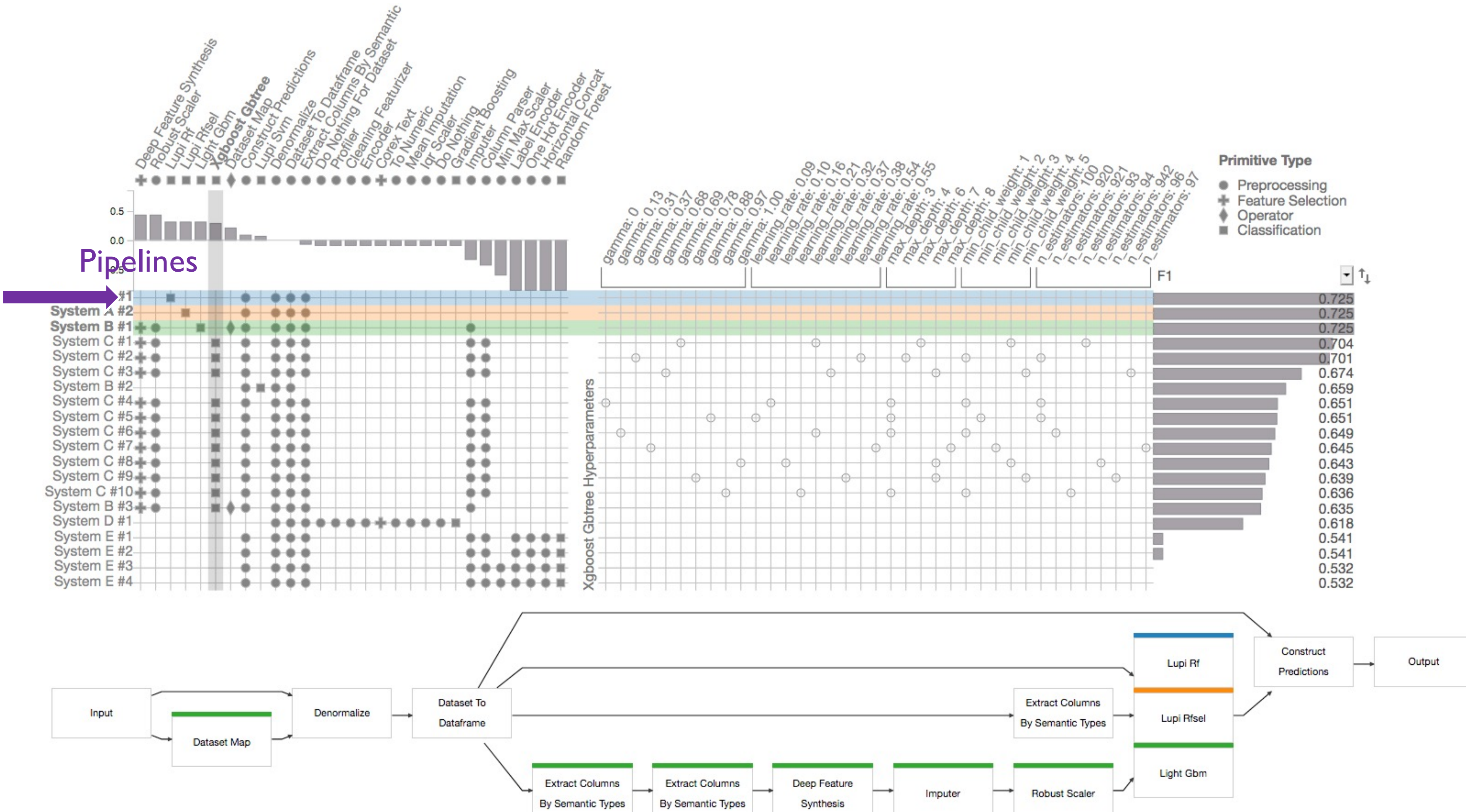
Pipeline Matrix

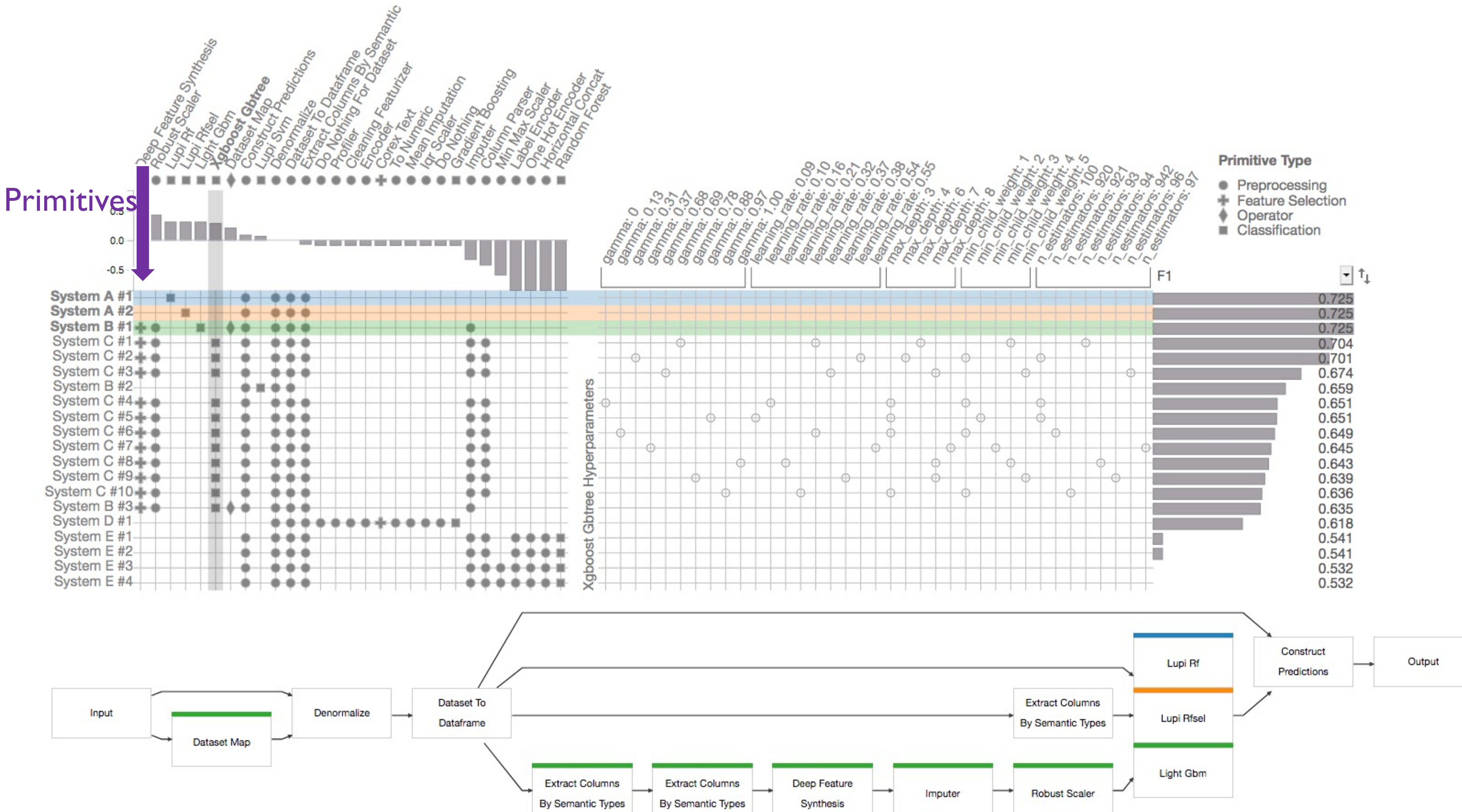




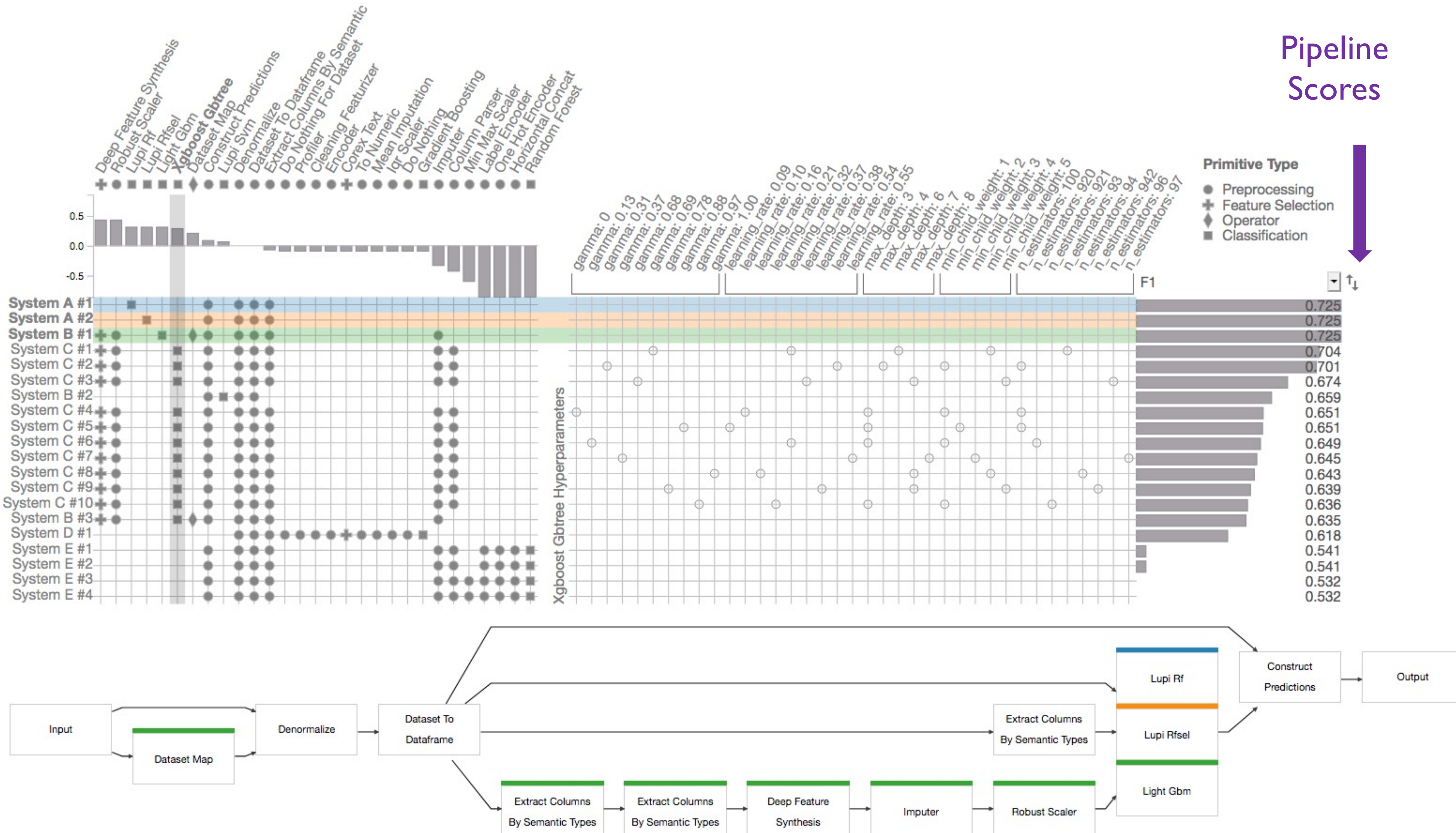
Pipeline Comparison View

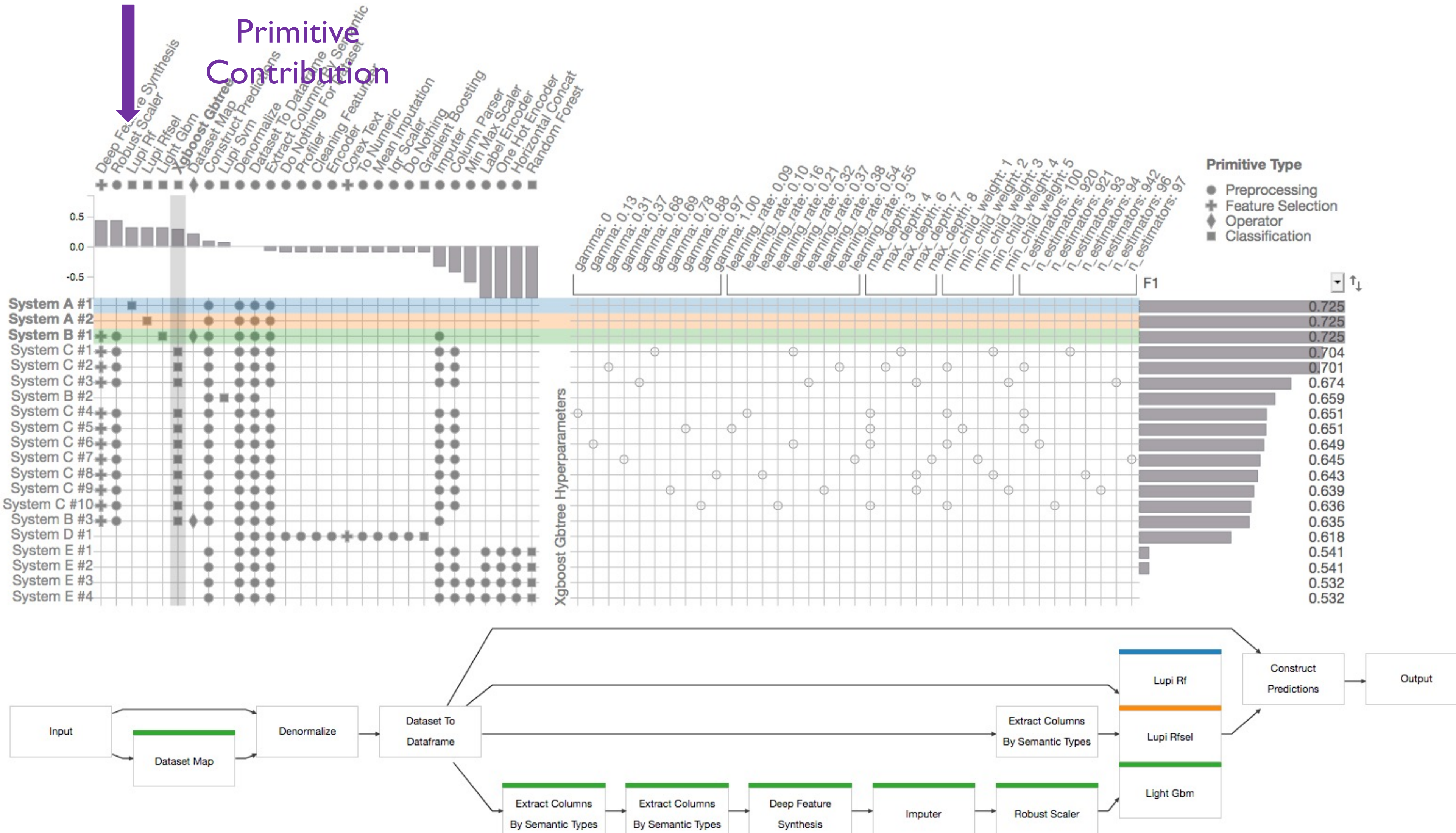


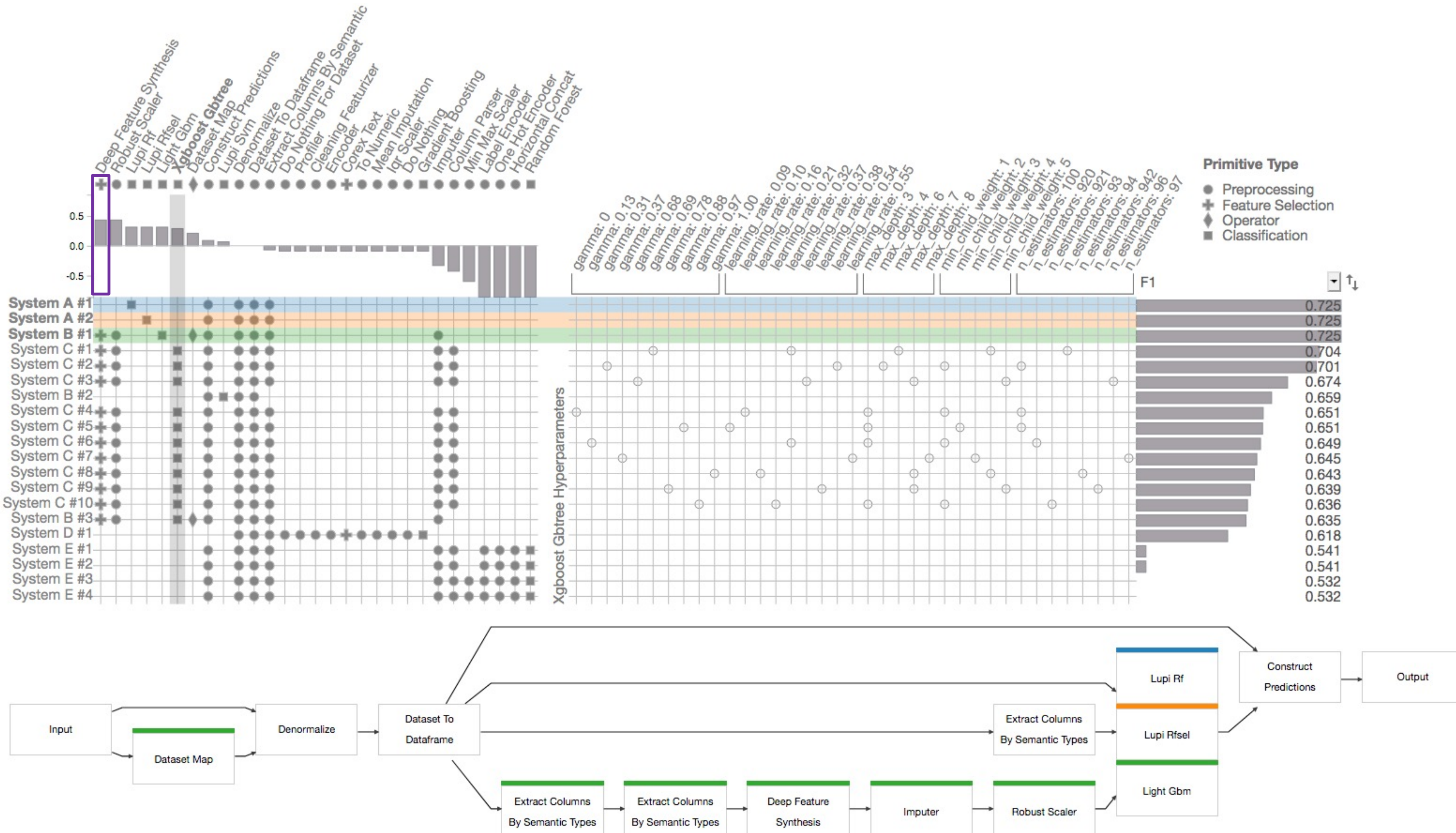

Pipelines
Deep Feature Synthesis
Robust Scaler
Lupi Rf
Lupi Rfsel
Light Gbm
Xgboost Gbtree
Dataset Map
Construct Predictions
Lupi Svm
Denormalize
Dataset To Dataframe
Extract Columns By Semantic
Do Nothing For Dataset
Profiler
Cleaning Featurizer
Encoder Text
Corex Text
To Numeric
Mean Imputation
Iqr Scaler
Do Nothing
Gradient Boosting
Imputer
Column Parser
Min Max Scaler
Label Encoder
One Hot Encoder
Horizontal Concat
Random Forest
0.5
0.0
-0.5
System A #1
System A #2
System B #1
System C #1
System C #2
System C #3
System B #2
System C #4
System C #5
System C #6
System C #7
System C #8
System C #9
System C #10
System B #3
System D #1
System E #1
System E #2
System E #3
System E #4
Xgboost Gbtree Hyperparameters
gamma: 0
gamma: 0.13
gamma: 0.31
gamma: 0.37
gamma: 0.68
gamma: 0.69
gamma: 0.78
gamma: 0.88
gamma: 0.97
gamma: 1.00
learning_rate: 0.09
learning_rate: 0.10
learning_rate: 0.16
learning_rate: 0.21
learning_rate: 0.32
learning_rate: 0.37
learning_rate: 0.38
learning_rate: 0.54
learning_rate: 0.55
max_depth: 3
max_depth: 4
max_depth: 6
max_depth: 7
max_depth: 8
min_child_weight: 1
min_child_weight: 2
min_child_weight: 3
min_child_weight: 4
min_child_weight: 5
n_estimators: 100
n_estimators: 920
n_estimators: 921
n_estimators: 93
n_estimators: 94
n_estimators: 942
n_estimators: 96
n_estimators: 97
F1
0.725
0.725
0.725
0.704
0.701
0.674
0.659
0.651
0.651
0.649
0.645
0.643
0.639
0.636
0.635
0.618
0.541
0.541
0.532
0.532
Primitive Type
Preprocessing
Feature Selection
Operator
Classification
Input
Dataset Map
Denormalize
Dataset To Dataframe
Extract Columns By Semantic Types
Extract Columns By Semantic Types
Deep Feature Synthesis
Imputer
Robust Scaler
Extract Columns By Semantic Types
Lupi Rf
Lupi Rfsel
Light Gbm
Construct Predictions
Output



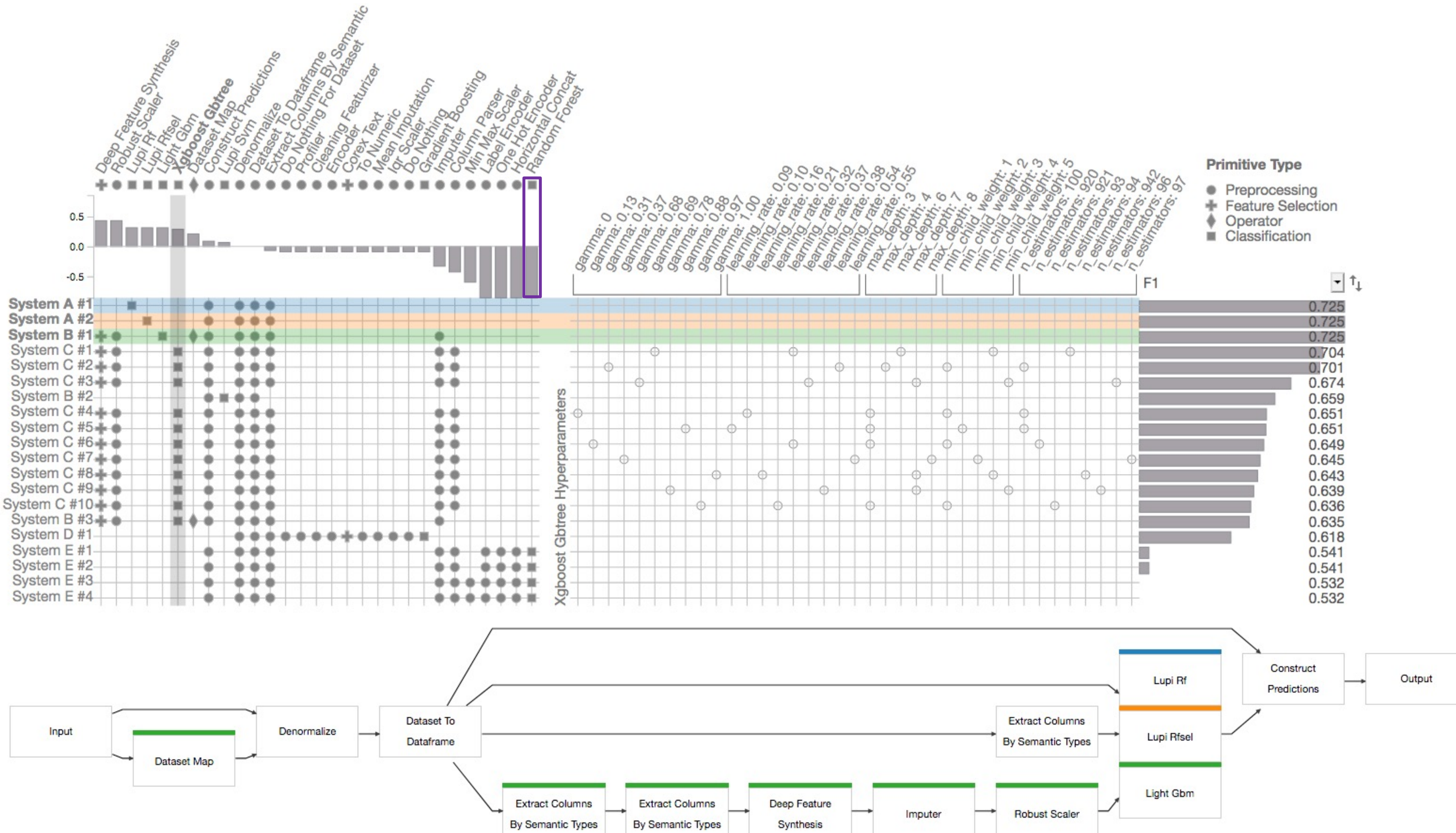
Primitives
Deep Feature Synthesis
Robust Scaler
Lupi Rf
Lupi Rfsel
Light Gbm
Xgboost Gbtree
Dataset Map
Construct Predictions
Lupi Svm
Denormalize
Dataset To Dataframe
Extract Columns By Semantic
Do Nothing For Dataset
Profiler
Cleaning Featurizer
Encoder Text
Corex Text
To Numeric
Mean Imputation
Iqr Scaler
Do Nothing
Gradient Boosting
Imputer
Column Parser
Min Max Scaler
Label Encoder
One Hot Encoder
Horizontal Concat
Random Forest
0.5
0.0
-0.5
System A #1
System A #2
System B #1
System C #1
System C #2
System C #3
System B #2
System C #4
System C #5
System C #6
System C #7
System C #8
System C #9
System C #10
System B #3
System D #1
System E #1
System E #2
System E #3
System E #4
Xgboost Gbtree Hyperparameters
gamma: 0
gamma: 0.13
gamma: 0.31
gamma: 0.37
gamma: 0.68
gamma: 0.69
gamma: 0.78
gamma: 0.88
gamma: 0.97
gamma: 1.00
learning_rate: 0.09
learning_rate: 0.10
learning_rate: 0.16
learning_rate: 0.21
learning_rate: 0.32
learning_rate: 0.37
learning_rate: 0.38
learning_rate: 0.54
learning_rate: 0.55
max_depth: 3
max_depth: 4
max_depth: 6
max_depth: 7
max_depth: 8
min_child_weight: 1
min_child_weight: 2
min_child_weight: 3
min_child_weight: 4
min_child_weight: 5
n_estimators: 100
n_estimators: 920
n_estimators: 921
n_estimators: 93
n_estimators: 94
n_estimators: 942
n_estimators: 96
n_estimators: 97
F1
0.725
0.725
0.725
0.704
0.701
0.674
0.659
0.651
0.651
0.649
0.645
0.643
0.639
0.636
0.635
0.618
0.541
0.541
0.532
0.532
Primitive Type
Preprocessing
Feature Selection
Operator
Classification
Input
Dataset Map
Denormalize
Dataset To Dataframe
Extract Columns By Semantic Types
Extract Columns By Semantic Types
Deep Feature Synthesis
Imputer
Robust Scaler
Extract Columns By Semantic Types
Lupi Rf
Lupi Rfsel
Light Gbm
Construct Predictions
Output


Pipeline Scores
Primitive Type
Preprocessing
Feature Selection
Operator
Classification
Deep Feature Synthesis
Robust Scaler
Lupi Rf
Lupi Rfsel
Light Gbm
Xgboost Gbtree
Dataset Map
Construct Predictions
Lupi Svm
Denormalize
Dataset To Dataframe
Extract Columns By Semantic
Do Nothing For Dataset
Profiler
Cleaning Featurizer
Encoder Text
Corex Text
To Numeric
Mean Imputation
Iqr Scaler
Do Nothing
Gradient Boosting
Imputer
Column Parser
Min Max Scaler
Label Encoder
One Hot Encoder
Horizontal Concat
Random Forest
0.5
0.0
-0.5
Xgboost Gbtree Hyperparameters
F1
System A #1 0.725
System A #2 0.725
System B #1 0.725
System C #1 0.704
System C #2 0.701
System C #3 0.674
System B #2 0.659
System C #4 0.651
System C #5 0.651
System C #6 0.649
System C #7 0.645
System C #8 0.643
System C #9 0.639
System C #10 0.636
System B #3 0.635
System D #1 0.618
System E #1 0.541
System E #2 0.541
System E #3 0.532
System E #4 0.532
Input
Dataset Map
Denormalize
Dataset To Dataframe
Extract Columns By Semantic Types
Deep Feature Synthesis
Imputer
Robust Scaler
Lupi Rf
Lupi Rfsel
Light Gbm
Construct Predictions
Output

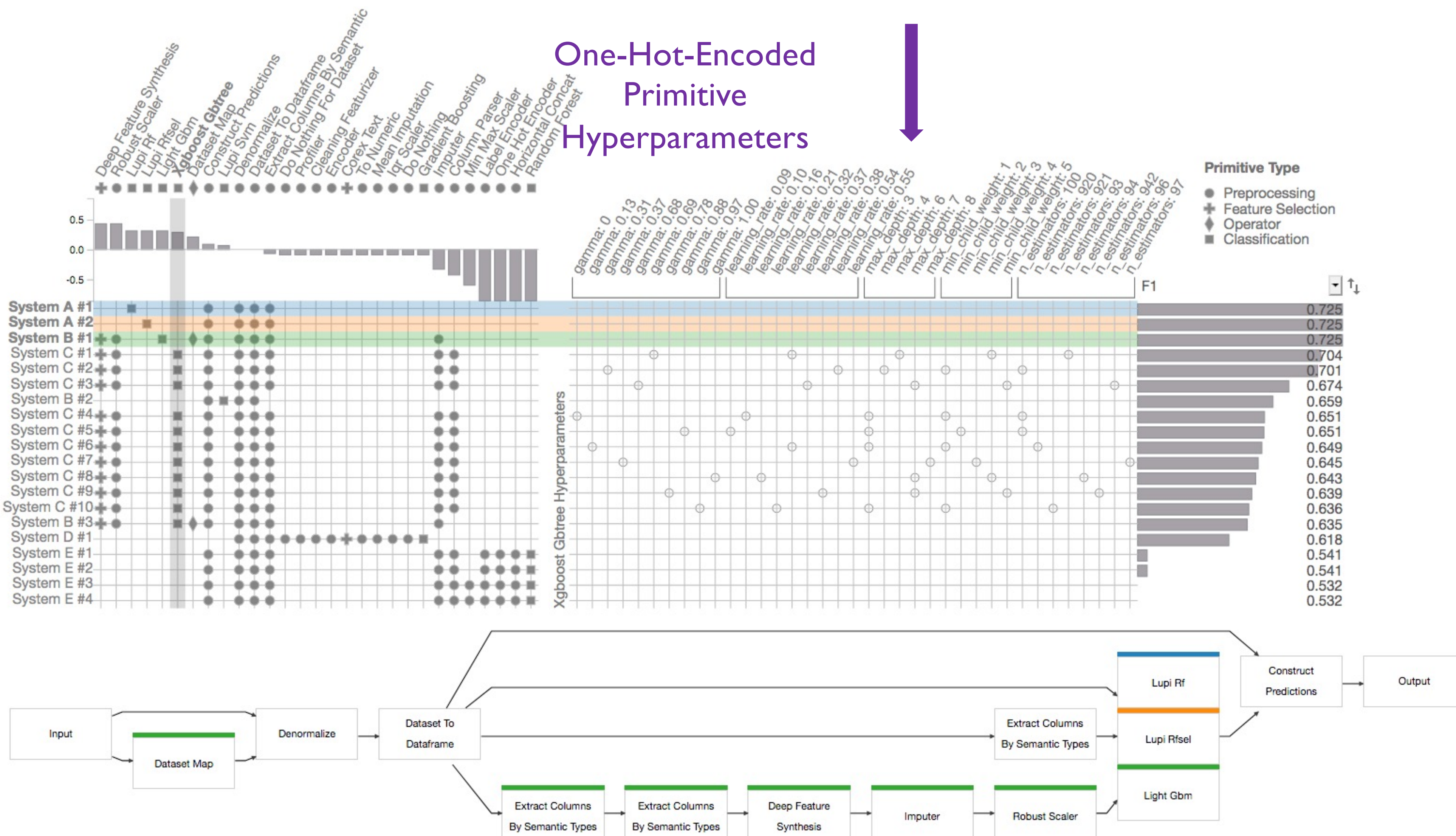

Primitive Contribution
Primitive Type
Preprocessing
Feature Selection
Operator
Classification
F1
Xgboost Gbtree Hyperparameters
System A #1 0.725
System A #2 0.725
System B #1 0.725
System C #1 0.704
System C #2 0.701
System C #3 0.674
System B #2 0.659
System C #4 0.651
System C #5 0.651
System C #6 0.649
System C #7 0.645
System C #8 0.643
System C #9 0.639
System C #10 0.636
System B #3 0.635
System D #1 0.618
System E #1 0.541
System E #2 0.541
System E #3 0.532
System E #4 0.532
Input
Dataset Map
Denormalize
Dataset To Dataframe
Extract Columns By Semantic Types
Deep Feature Synthesis
Imputer
Robust Scaler
Lupi Rf
Lupi Rfsel
Light Gbm
Construct Predictions
Output



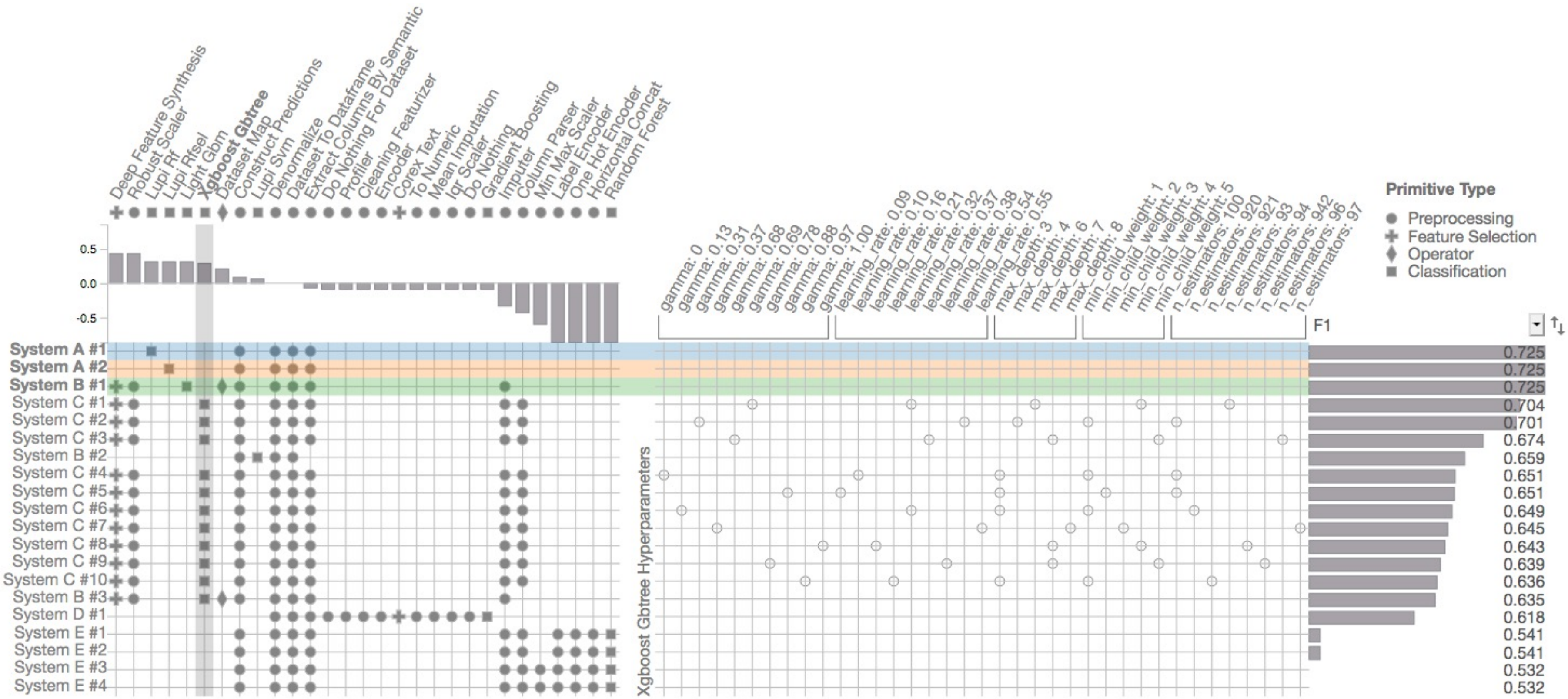
Deep Feature Synthesis
Robust Scaler
Lupi Rf
Lupi Rfsel
Light Gbm
Xgboost Gbtree
Dataset Map
Construct Predictions
Lupi Svm
Denormalize
Dataset To Dataframe
Extract Columns By Semantic
Do Nothing For Dataset
Profiler
Cleaning Featurizer
Encoder Text
Corex Text
To Numeric
Mean Imputation
Iqr Scaler
Do Nothing
Gradient Boosting
Imputer
Column Parser
Min Max Scaler
Label Encoder
One Hot Encoder
Horizontal Concat
Random Forest
Primitive Type
Preprocessing
Feature Selection
Operator
Classification
F1
Xgboost Gbtree Hyperparameters
System A #1
System A #2
System B #1
System C #1
System C #2
System C #3
System B #2
System C #4
System C #5
System C #6
System C #7
System C #8
System C #9
System C #10
System B #3
System D #1
System E #1
System E #2
System E #3
System E #4
Input
Dataset Map
Denormalize
Dataset To Dataframe
Extract Columns By Semantic Types
Deep Feature Synthesis
Imputer
Robust Scaler
Lupi Rf
Lupi Rfsel
Light Gbm
Construct Predictions
Output



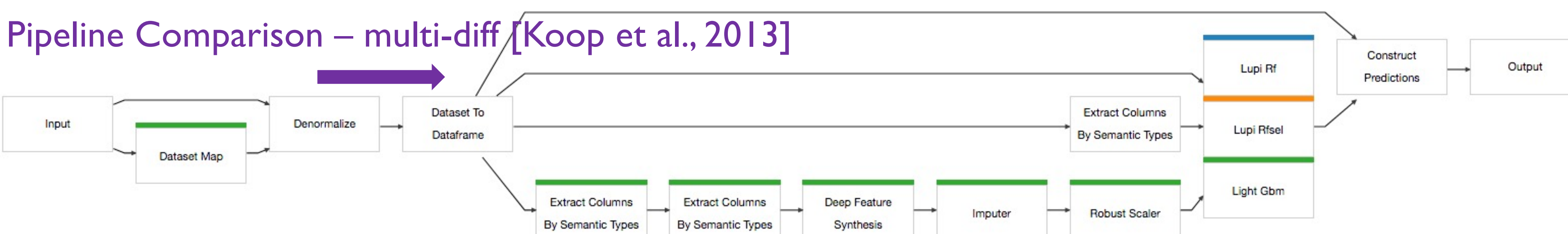
Deep Feature Synthesis
Robust Scaler
Lupi Rf
Lupi Rfsel
Light Gbm
Xgboost Gbtree
Dataset Map
Construct Predictions
Lupi Svm
Denormalize
Dataset To Dataframe
Extract Columns By Semantic
Do Nothing For Dataset
Profiler
Cleaning Featurizer
Encoder Text
Corex Text
To Numeric
Mean Imputation
Iqr Scaler
Do Nothing
Gradient Boosting
Imputer
Column Parser
Min Max Scaler
Label Encoder
One Hot Encoder
Horizontal Concat
Random Forest
0.5
0.0
-0.5
System A #1
System A #2
System B #1
System C #1
System C #2
System C #3
System B #2
System C #4
System C #5
System C #6
System C #7
System C #8
System C #9
System C #10
System B #3
System D #1
System E #1
System E #2
System E #3
System E #4
Xgboost Gbtree Hyperparameters
gamma: 0
gamma: 0.13
gamma: 0.31
gamma: 0.37
gamma: 0.68
gamma: 0.69
gamma: 0.78
gamma: 0.88
gamma: 0.97
gamma: 1.00
learning_rate: 0.09
learning_rate: 0.10
learning_rate: 0.16
learning_rate: 0.21
learning_rate: 0.32
learning_rate: 0.37
learning_rate: 0.38
learning_rate: 0.54
learning_rate: 0.55
max_depth: 3
max_depth: 4
max_depth: 6
max_depth: 7
max_depth: 8
min_child_weight: 1
min_child_weight: 2
min_child_weight: 3
min_child_weight: 4
min_child_weight: 5
n_estimators: 100
n_estimators: 920
n_estimators: 921
n_estimators: 93
n_estimators: 94
n_estimators: 942
n_estimators: 96
n_estimators: 97
F1
0.725
0.725
0.725
0.704
0.701
0.674
0.659
0.651
0.651
0.649
0.645
0.643
0.639
0.636
0.635
0.618
0.541
0.541
0.532
0.532
Primitive Type
Preprocessing
Feature Selection
Operator
Classification
Input
Dataset Map
Denormalize
Dataset To Dataframe
Extract Columns By Semantic Types
Extract Columns By Semantic Types
Deep Feature Synthesis
Imputer
Robust Scaler
Extract Columns By Semantic Types
Lupi Rf
Lupi Rfsel
Light Gbm
Construct Predictions
Output



One-Hot-Encoded
Primitive
Hyperparameters
Deep Feature Synthesis
Robust Scaler
Lupi Rf
Lupi Rfsel
Light Gbm
Xgboost Gbtree
Dataset Map
Construct Predictions
Lupi Svm
Denormalize
Dataset To Dataframe
Extract Columns By Semantic
Do Nothing For Dataset
Profiler
Cleaning Featurizer
Encoder Text
Corex Text
To Numeric
Mean Imputation
Iqr Scaler
Do Nothing
Gradient Boosting
Imputer
Column Parser
Min Max Scaler
Label Encoder
One Hot Encoder
Horizontal Concat
Random Forest
0.5
0.0
-0.5
System A #1
System A #2
System B #1
System C #1
System C #2
System C #3
System B #2
System C #4
System C #5
System C #6
System C #7
System C #8
System C #9
System C #10
System B #3
System D #1
System E #1
System E #2
System E #3
System E #4
Xgboost Gbtree Hyperparameters
gamma: 0
gamma: 0.13
gamma: 0.31
gamma: 0.37
gamma: 0.68
gamma: 0.69
gamma: 0.78
gamma: 0.88
gamma: 0.97
gamma: 1.00
learning_rate: 0.09
learning_rate: 0.10
learning_rate: 0.16
learning_rate: 0.21
learning_rate: 0.32
learning_rate: 0.37
learning_rate: 0.38
learning_rate: 0.54
learning_rate: 0.55
max_depth: 3
max_depth: 4
max_depth: 6
max_depth: 7
max_depth: 8
min_child_weight: 1
min_child_weight: 2
min_child_weight: 3
min_child_weight: 4
min_child_weight: 5
n_estimators: 100
n_estimators: 920
n_estimators: 921
n_estimators: 93
n_estimators: 94
n_estimators: 942
n_estimators: 96
n_estimators: 97
Primitive Type
Preprocessing
Feature Selection
Operator
Classification
F1
0.725
0.725
0.725
0.704
0.701
0.674
0.659
0.651
0.651
0.649
0.645
0.643
0.639
0.636
0.635
0.618
0.541
0.541
0.532
0.532
Input
Dataset Map
Denormalize
Dataset To Dataframe
Extract Columns By Semantic Types
Extract Columns By Semantic Types
Extract Columns By Semantic Types
Deep Feature Synthesis
Imputer
Robust Scaler
Lupi Rf
Lupi Rfsel
Light Gbm
Construct Predictions
Output



Pipeline Comparison – multi-diff [Koop et al., 2013]



Expert Feedback

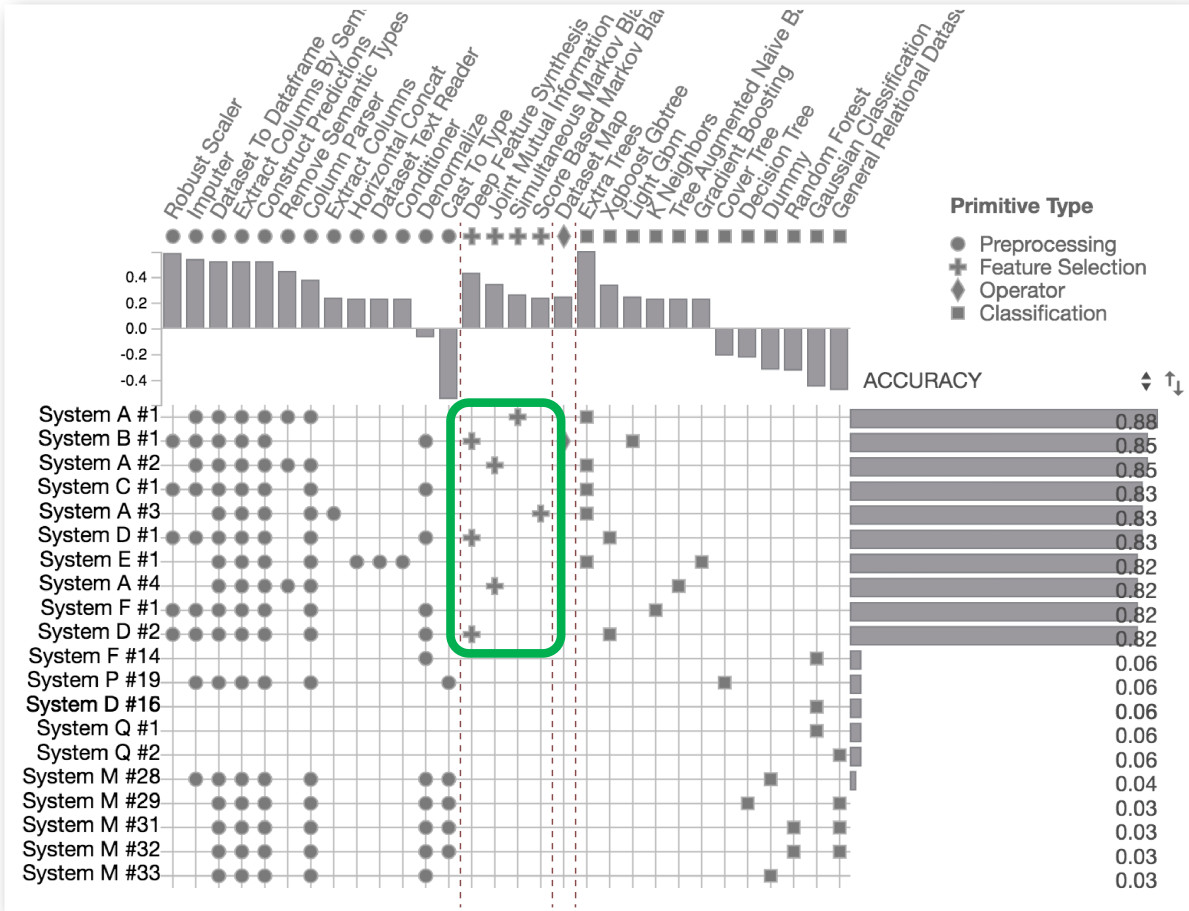
- System was adopted by D3M members
- We conducted think aloud interviews with 6 D3M Data Scientists
- The experts liked the tool and used it to gain insights and improve their systems:
  - Discovered useful primitives
  - Assessed primitive correctness
  - Compared hyperparameter search strategies
  - Understood search strategies of AutoML systems – by reverse engineering…

Impact of Primitives on Scores

Libras Move Classification Dataset

Feature selection primitives have a big impact in this classification problem.

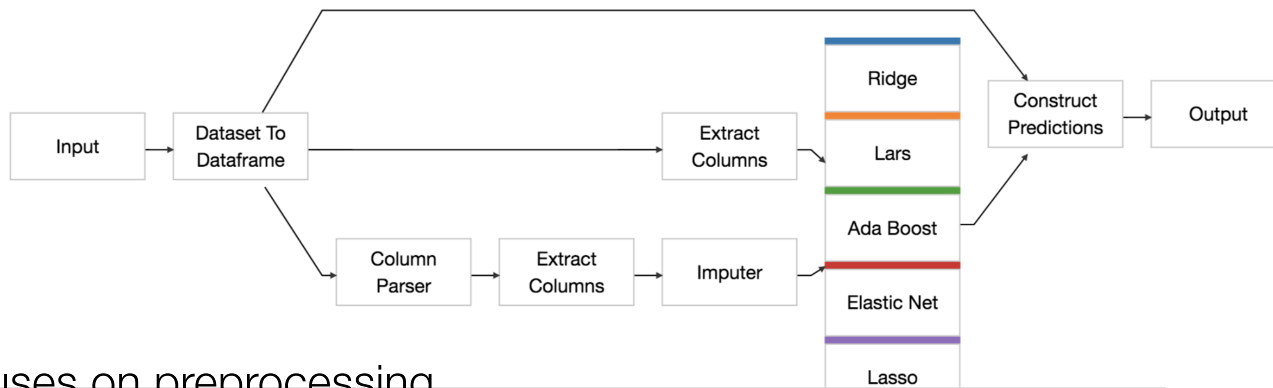
Actionable Insight:
Extended the AutoML search to consider these primitives. Improvement in accuracy, from 0.79 to 0.88.



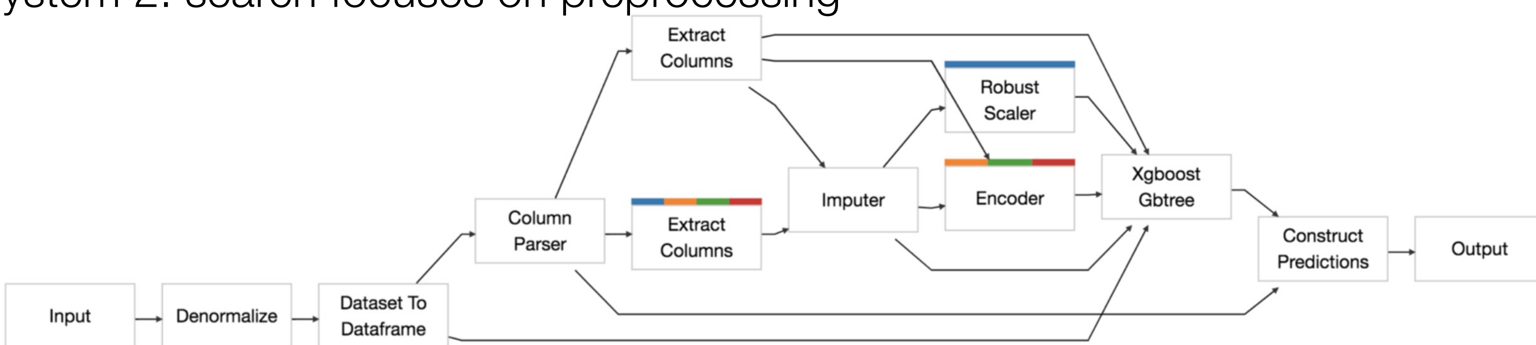
Reverse Engineering AutoML Search

CPS Wages Regression Dataset

AutoML System 1: search focuses on estimators



AutoML System 2: search focuses on preprocessing



Understanding and Customizing Pipelines

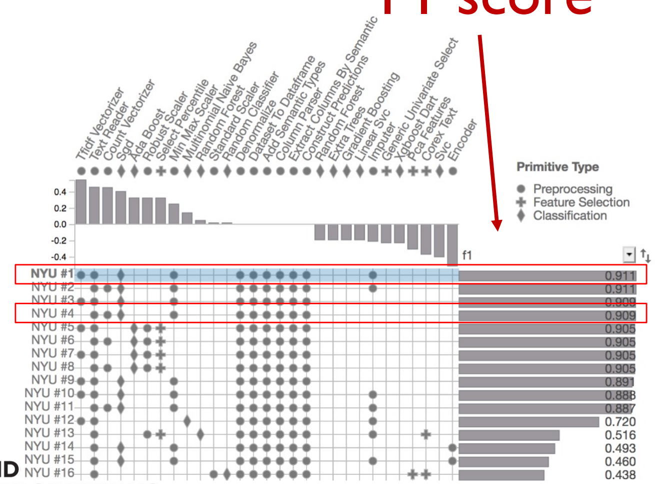
Data: Articles describing events involving terrorist activities

Goal: identify articles that describe attacks involving explosions

AlphaD3M generates high-quality pipelines

- Best pipeline has F1 score of 0.911 and execution time of 811 secs (~14 mins)
- Other pipelines have similar score and faster execution times. But why?

F1 score

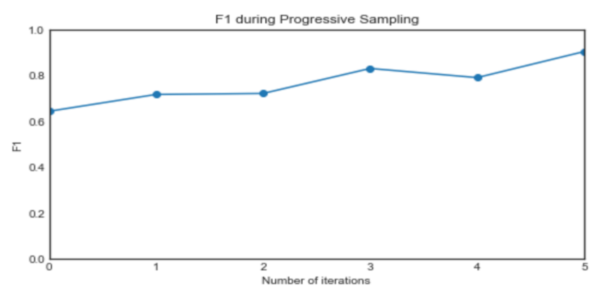


Execution time

Reducing Execution Time

- Long running times can be a problem for pipelines that need to be deployed and used in production; and also limits the AutoML search
- We implemented a new method that combines progressive sampling and active learning to reduce the size of the training dataset

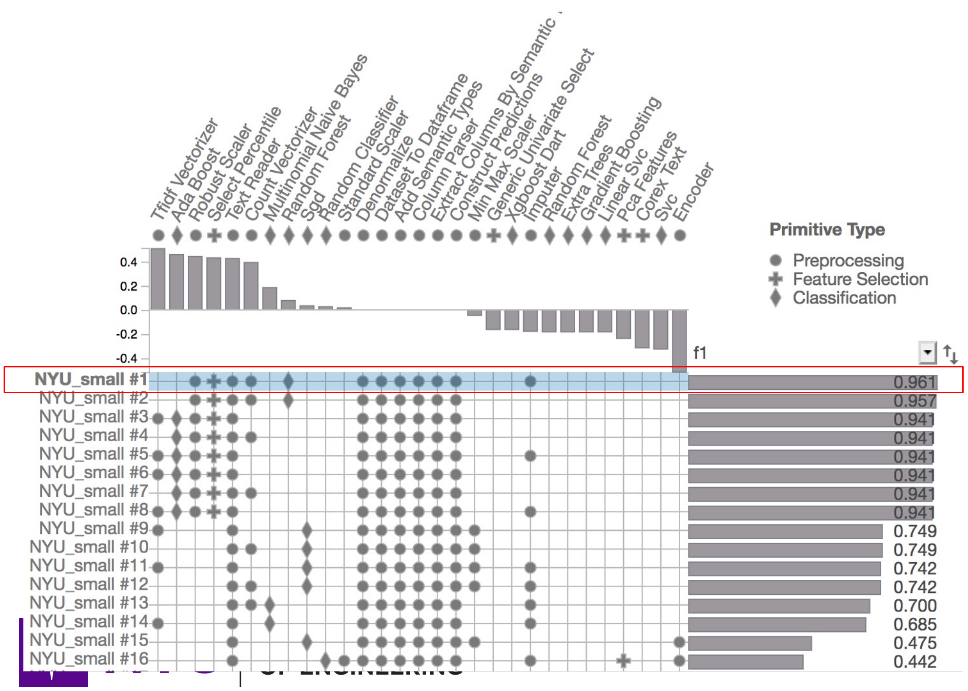
```
In [11]: reduced_dataset = progressive_sampling(initial_dataset, pool_dataset, test_dataset)

F1 score (dataset size=100): 0.6438
F1 score (dataset size=200): 0.7171
F1 score (dataset size=400): 0.7214
F1 score (dataset size=800): 0.8303
F1 score (dataset size=1600): 0.7903
F1 score (dataset size=3200): 0.9039
Threshold reached!
```

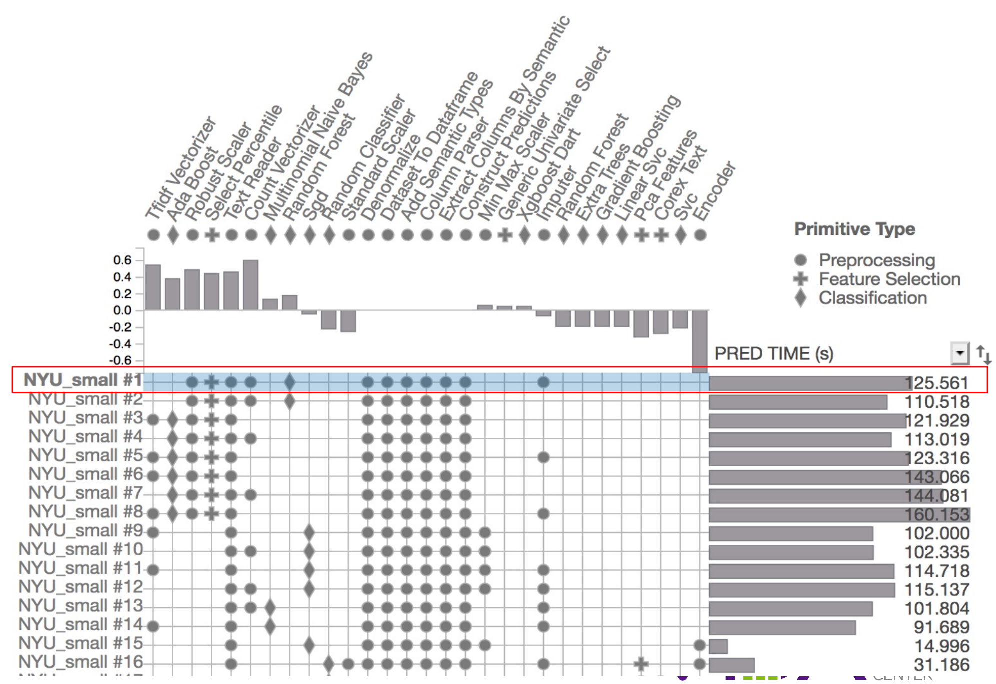


Integrate new primitives to address problem-specific challenges

Improved Results after Sampling

- Best performance improved from 0.911 to **0.961** of F1
- Best execution time was reduced from 811 to **125** secs





Improving Models through Data Augmentation

Scalable Storage

PostgreSQL
MINIO
Google Cloud Storage
Google BigQuery
amazon S3
snowflake
Azure Data lake

Open Data

NYC OpenData
DATA.GOV
Stadt Zürich Open Data
GOVDATA Das Datenportal für Deutschland
THE WORLD BANK IBRD • IDA | Data
NIH NATIONAL CANCER INSTITUTE CIP Cancer Imaging Program
NIH NATIONAL CANCER INSTITUTE Genomic Data Commons
About the GDC
The Sloan Digital Sky Survey: Mapping the Univ

Estimate… 200-300 million datasets
on the Web
[Noy 2023]

Data abundance creates new opportunities to improve ML models

Taxi Demand Prediction

Taxi Trips Data

NYC Taxi & Limousine Commission

Taxi

pickup_datetime	LocationID	n. trips	…
2017-01-01 00:00:00	4	136	
2017-01-01 01:00:00	7	78	
2017-02-01 10:00:00	12	189	
2017-01-10 13:00:00	23	56	
2017-04-15 22:00:00	17	4	

↓

Random Forest Regressor

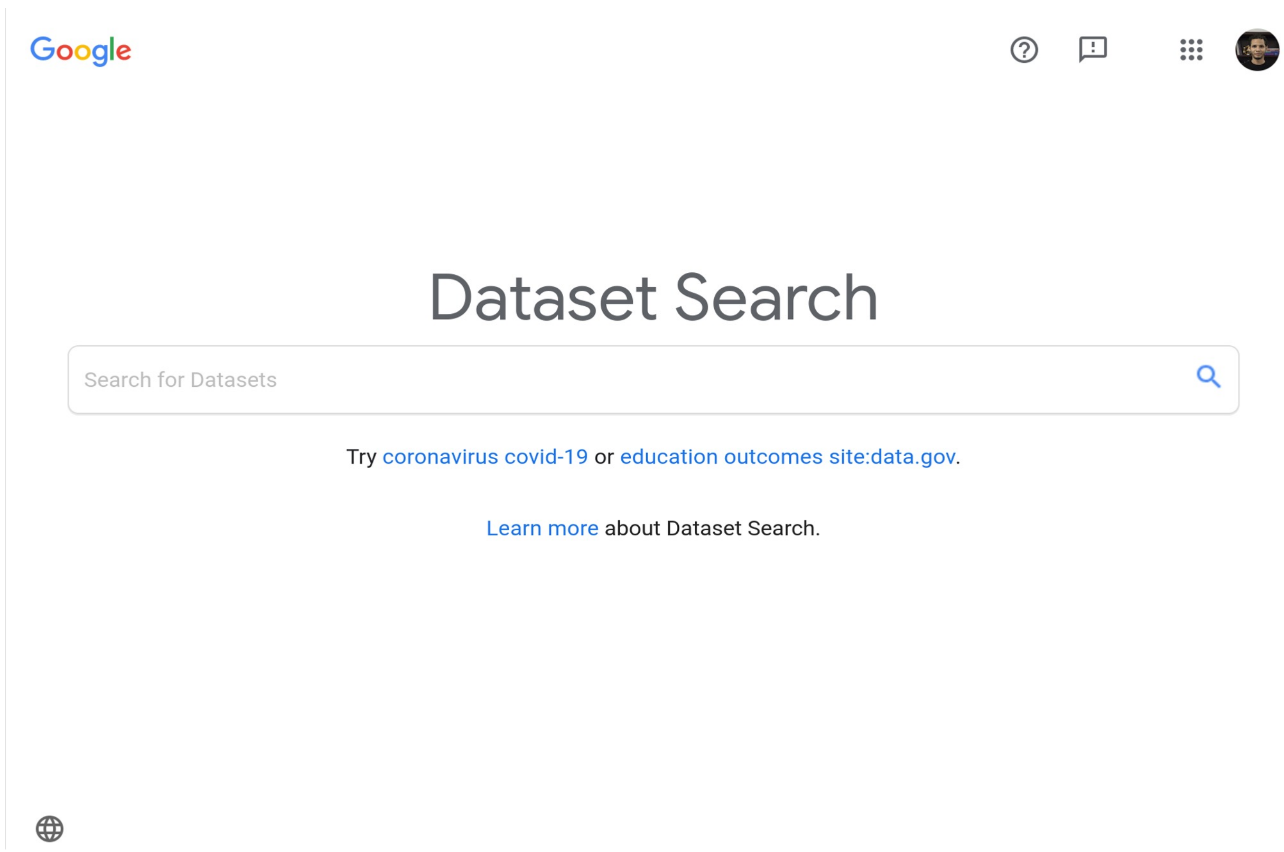
↓

MAE: 66.67

Can we find additional features to improve this model?

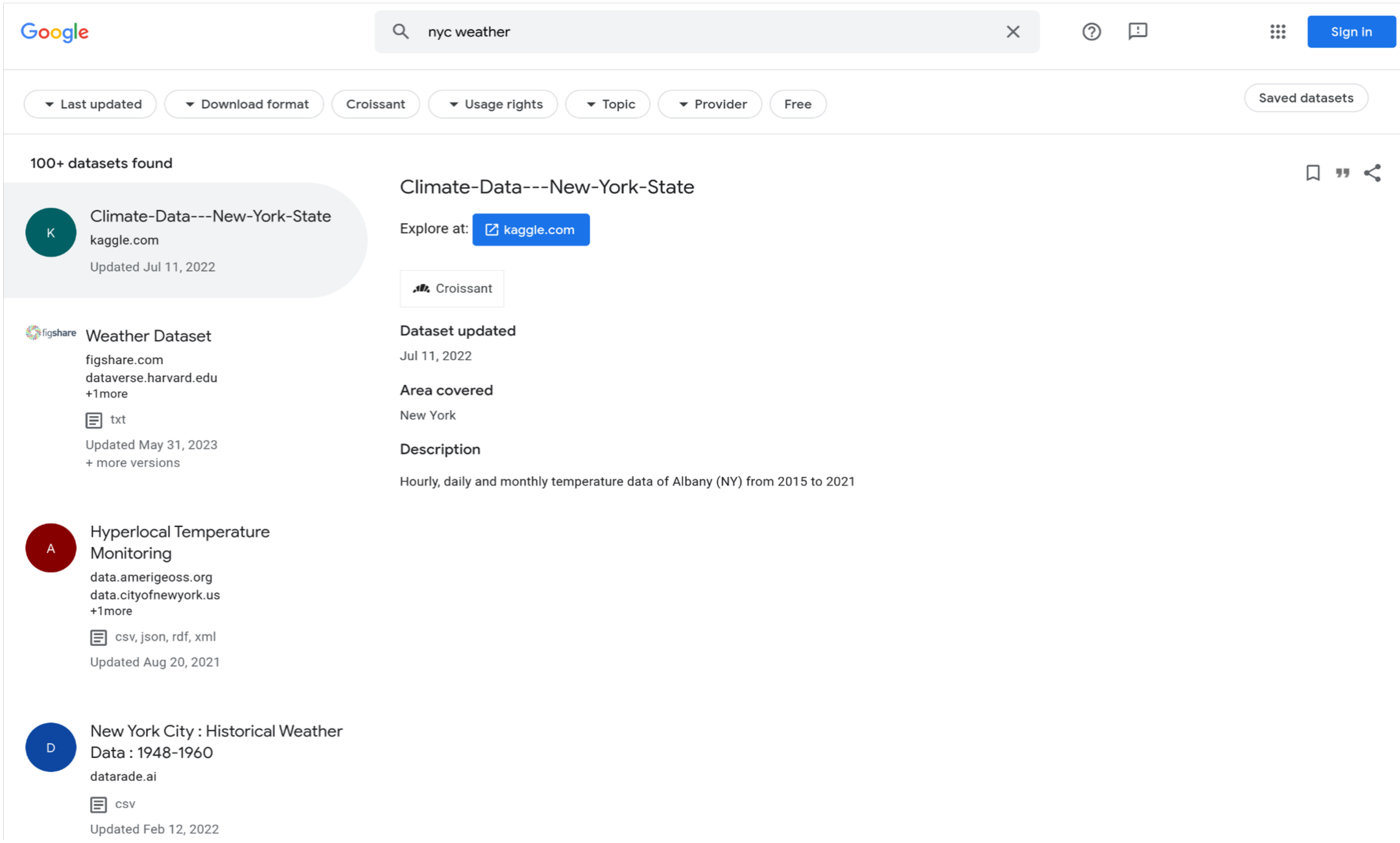
NYU TANDON SCHOOL OF ENGINEERING

VIDA VISUALIZATION IMAGING AND DATA ANALYSIS CENTER



NYU | TANDON SCHOOL OF ENGINEERING

VIDA VISUALIZATION IMAGING AND DATA ANALYSIS CENTER



VISUALIZATION
IMAGING AND
DATA ANALYSIS
CENTER

Taxi Demand Prediction

Taxi Trips Data

+

Weather Indicators

Date	Temperature				HDD	CDD	Precipitation	New Snow	Snow Depth
	Maximum	Minimum	Average	Departure					
2017-01-01	48	40	44.0	8.8	21	0	T	0.0	0
2017-01-02	41	37	39.0	4.0	26	0	0.21	T	0
2017-01-03	43	39	41.0	6.2	24	0	0.58	0.0	0
2017-01-04	52	34	43.0	8.3	22	0	0.00	0.0	0
2017-01-05	34	27	30.5	-4.0	34	0	0.00	0.0	0
2017-01-06	33	25	29.0	-5.4	36	0	0.05	1.2	1
2017-01-07	26	20	23.0	-11.2	42	0	0.32	5.1	T
2017-01-08	25	16	20.5	-13.6	44	0	0.00	0.0	4
2017-01-09	23	14	18.5	-15.4	46	0	0.00	0.0	3
2017-01-10	46	21	33.5	-0.3	31	0	0.00	0.0	3
2017-01-11	52	42	47.0	13.3	18	0	0.52	0.0	0
2017-01-12	66	47	56.5	22.9	8	0	0.05	0.0	0
2017-01-13	62	32	47.0	13.5	18	0	0.00	0.0	0

datetime	...	precip	temp
2017-01-01 00:00:00		0.0	7.2
2017-01-01 01:00:00		0.0	7.2
2017-02-01 10:00:00		1.0	5.0
2017-01-10 13:00:00		0.0	-1.2

Random Forest Regressor

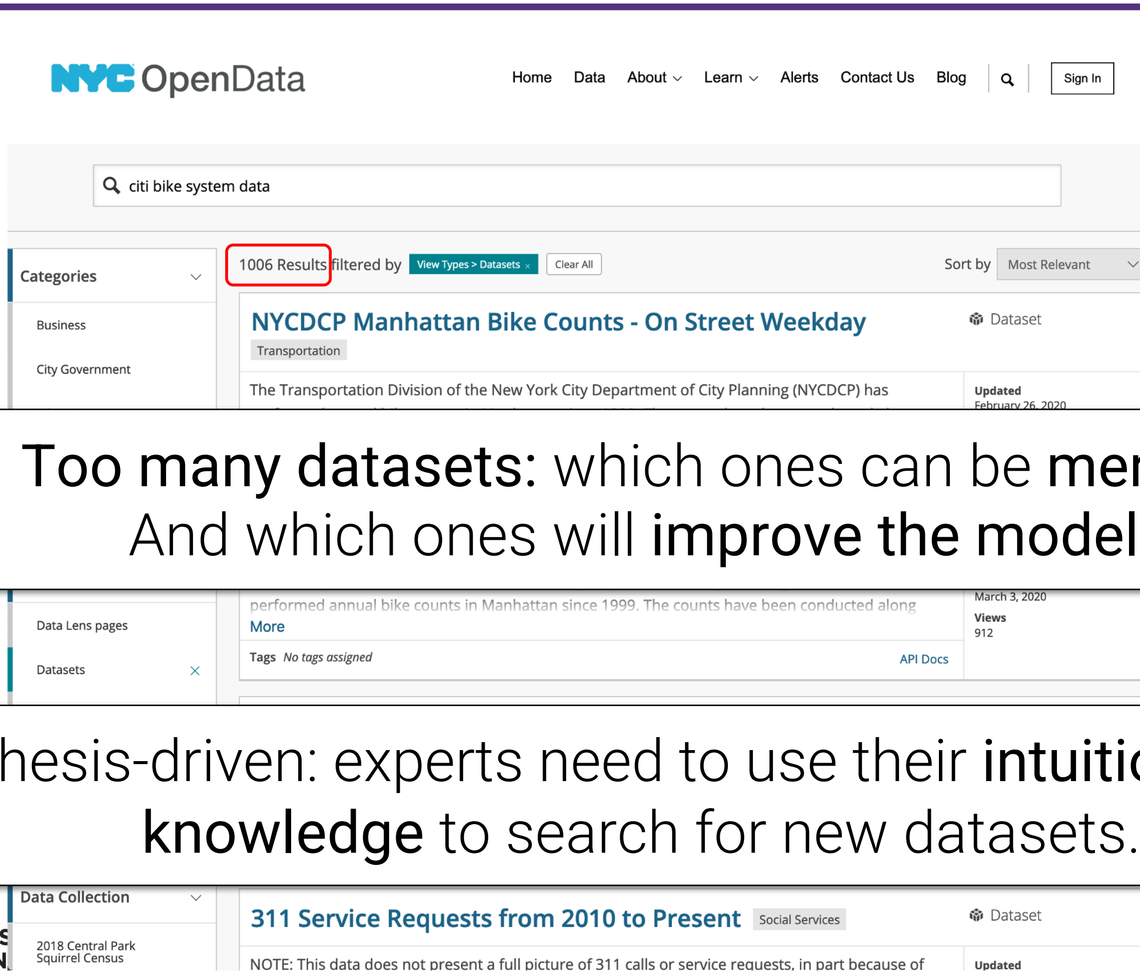
MAE: 39.30

41% improvement!

NYU TANDON SCHOOL OF ENGINEERING

VIDA VISUALIZATION IMAGING AND DATA ANALYSIS CENTER

Discovering Relevant Data: Challenges



Too many datasets: which ones can be **merged**?
And which ones will **improve the model**?

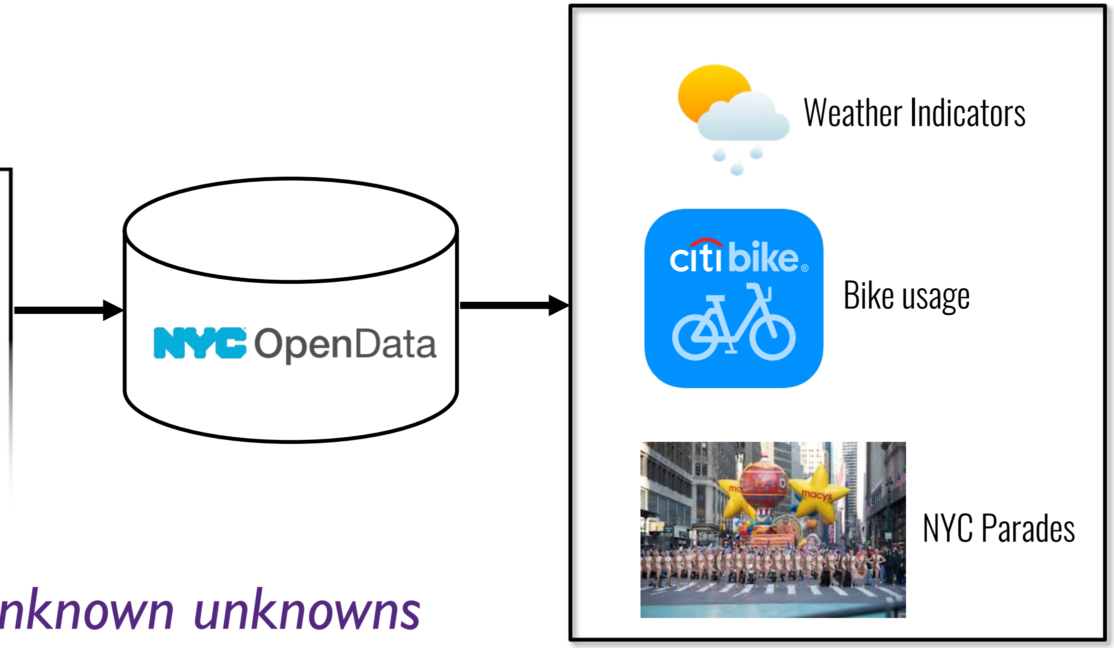
Hypothesis-driven: experts need to use their **intuition** and **prior knowledge** to search for new datasets.

Dataset Discovery Queries

- Query is a dataset → return *related datasets*
  - Related = joinable and improve the model

Taxi

pickup_datetime	LocationID	n. trips	…
2017-01-01 00:00:00	4	136	
2017-01-01 01:00:00	7	78	
2017-02-01 10:00:00	12	189	
2017-01-10 13:00:00	23	56	
2017-04-15 22:00:00	17	4	



Goal: discover *unknown unknowns*

NYU TANDON SCHOOL OF ENGINEERING

VIDA VISUALIZATION IMAGING AND DATA ANALYSIS CENTER

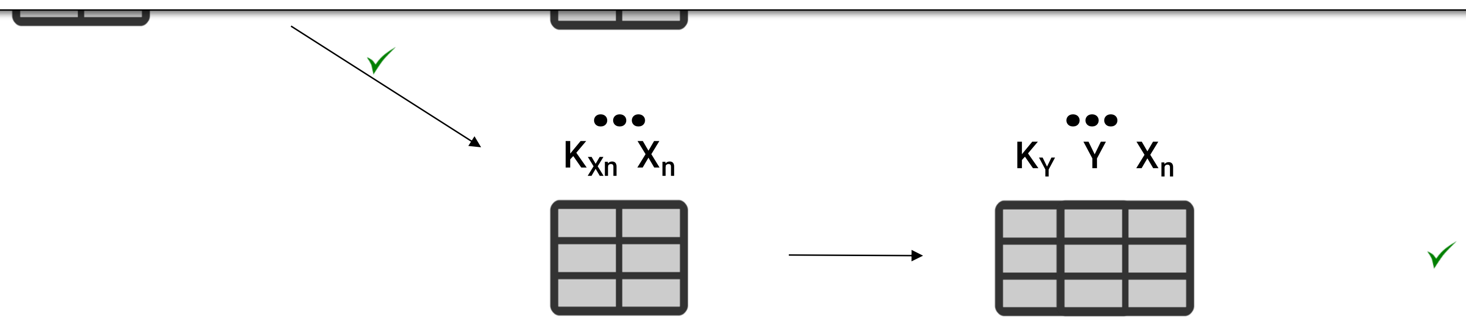
Relational Data Augmentation

Data lake with many candidate tables

Augmented table after a left join

Improves model performance?

A naïve approach is expensive for large collections, and can be wasteful as many tables can be joinable but do not lead to model improvement



Find joinable tables [Zhu et al., SIGMOD 2019; Zhu et al., VLDB 2016; Fernandez et al., ICDE 2019; Yang et al., ICDE 2019]

Join-Correlation Queries

- A data-driven approach for data discovery: find datasets that are *joinable and correlated*
- Useful features are typically correlated with the target variable
  - This has been used in many feature selection algorithms

Finding correlated data in large table collections may help to "explain" or "predict" other variables of interest

Taxi demand model: Find all datasets that join with the ***NYC taxi data*** *and contain an attribute that is correlated with the target variable* ***number of trips***

NYU TANDON SCHOOL OF ENGINEERING

[Santos et al., ACM SIGMOD 2021, IEEE ICDE 2023]

VIDA VISUALIZATION IMAGING AND DATA ANALYSIS CENTER

Join-Correlation Queries

Information need:
How can we **efficiently** find variables which help **predict** a target variable in **large-scale** dataset collections?

Finding correlated data in large table collections may help to "explain" or "predict" other variables of interest

*Taxi demand model: Find all datasets that join with the **NYC taxi data** and contain an attribute that is correlated with the target variable **number of trips***

[Santos et al., ACM SIGMOD 2021, IEEE ICDE 2023, ICDE 2024]

Join-Correlation Queries

Problem Definition:

Given a query table $T_Q = (K_Q, Q)$ where:

1) K_Q is a join column
2) Q is a target column

Find the top-k tables $T_C = (K_C, C)$ in a table collection such that:

1) T_C is joinable with T_Q on K_Q
2) T_C contains a column C that has a strong correlation with Q

[Santos et al., SIGMOD 2021]

Our Approach: Use Sketches to Estimate Correlation

- Idea: Reduce input size and derive approximate results
- Challenge: need to estimate **post-join** correlation of independent tables

$\mathcal{T}_X$

K_X	X
2021-01	6.0
2021-02	4.0
2021-03	2.0
2021-04	3.0
2021-05	0.5
2021-06	4.0
2021-07	2.0

$\mathcal{T}_Y$

K_Y	Y
2021-01	5.5
2021-01	4.5
2021-02	3.9
2021-02	2.0
2021-03	4.0
2021-03	1.0
2021-04	4.0

$\mathcal{T}_{X \bowtie Y}$

$K_{X \bowtie Y}$	$X_{X \bowtie Y}$	$Y_{X \bowtie Y}$
2021-04	3.0	4.0
2021-03	2.0	2.5
2021-02	4.0	3.0
2021-01	6.0	5.0

r = 0.81

Compute correlation between X and Y

- Randomly sampling column vectors does not yield valid correlations
- Sampling rows randomly does not work either!

NYU TANDON SCHOOL OF ENGINEERING

Values need to be aligned using the join key

VIDA VISUALIZATION IMAGING AND DATA ANALYSIS CENTER

Correlation Sketches

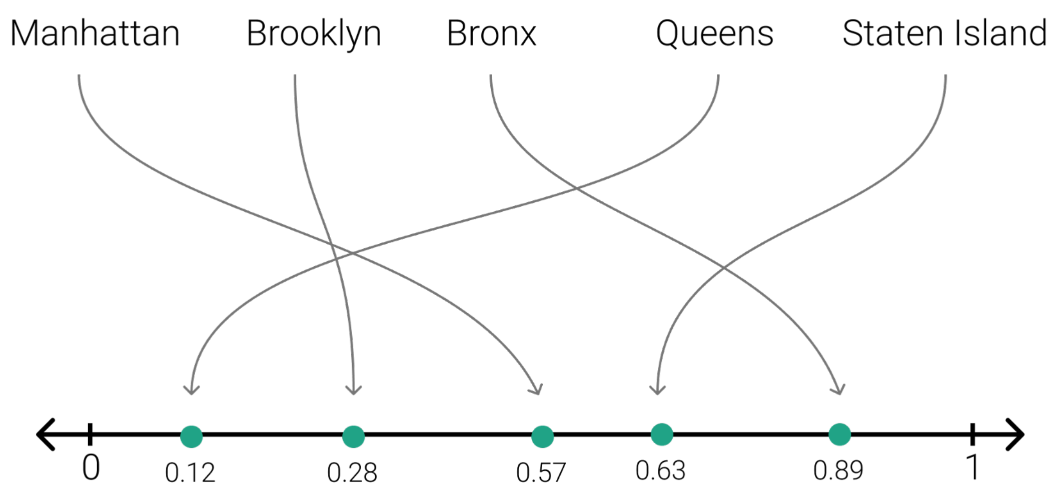
- Build sketches to estimate join-correlation
  - Use a **hashing function** to create a **data sketch** for each table in a collection – create *coordinated samples*
  - Given two sketches (X,Y), recover a ***uniform random sample*** of $T_{X \bowtie Y}$ without computing the full join
  - Apply *any* correlation estimator over the sketch join
- Two-steps:
  1. Discover the top-k most joinable tables using index for **fast joinable table retrieval**
  2. Perform join-correlation estimate at query time and re-rank candidates

  → Join-correlations are efficiently approximated using sketches

Extension of KMV sketches [Beyer et al., SIGMOD 2007]

Coordinated Sampling via Minwise Hashing

- Use hashing functions to create a data sketch for each table
- Select rows based on **minimum unit hash values** of $h_u(k)$
  - $L_{\langle K,X\rangle} = \{ \langle h(k), x_k \rangle \}$ with *n* minimum values of $h_u(k)$

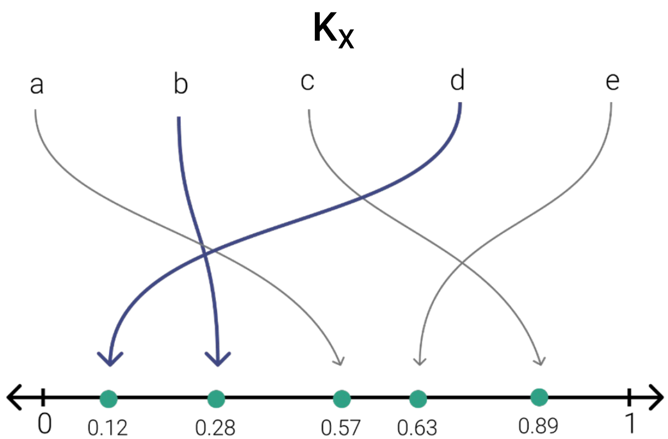


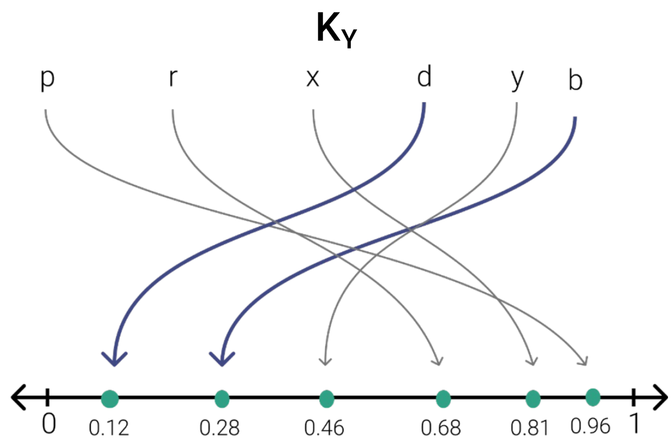
NYU TANDON SCHOOL OF ENGINEERING

VIDA VISUALIZATION IMAGING AND DATA ANALYSIS CENTER

Minwise Hashing Introduces Key Dependence

If a key is sampled from K_X, then it is more likely to be sampled from K_Y





$n=3 \rightarrow S_X = \{b, d, a\}$

$n=3 \rightarrow S_Y = \{b, d, y\}$

CSK Sketch Construction

hash join keys

map to [0,1]

K	X
2021-01	6.0
2021-02	4.0
2021-03	2.0
2021-04	3.0
2021-05	0.5
2021-06	4.0
2021-07	2.0

Apply hash to keys $\Longrightarrow$

h(k)	$h_u(k)$	x
4da33cf5	0.34	6.0
84c46s9d	0.90	4.0
bac52e98	0.48	2.0
26f79756	0.47	3.0
9d4fa4b6	0.83	0.5
73sc3d34	0.69	4.0
16dab449	0.34	2.0

keep X values

Select rows with minimum unit hash values $\Longrightarrow$

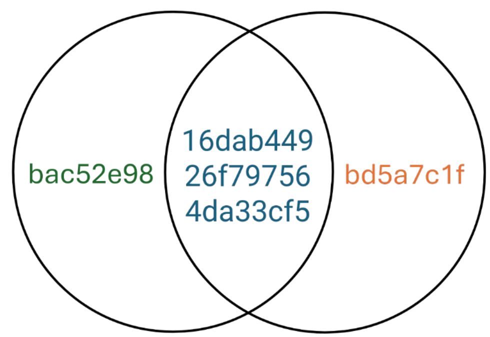
h(k)	$h_u(k)$	x
bac52e98	0.48	2.0
16dab449	0.34	2.0
26f79756	0.47	3.0
4da33cf5	0.34	6.0

Finding Joinable Tables

- Build an index of sketches' hash keys for **joinable table retrieval**

h(k)	$h_u(k)$	x
bac52e98	0.48	2.0
16dab449	0.34	2.0
26f79756	0.47	3.0
4da33cf5	0.34	6.0

h(k)	$h_u(k)$	y
bd5a7c1f	0.89	3.0
16dab449	0.34	2.5
26f79756	0.47	4.0
4da33cf5	0.34	5.0



Overlap = 3

Set Overlap Search:
Find the tables with highest overlap of hashed join keys

NYU TANDON SCHOOL OF ENGINEERING

VIDA VISUALIZATION IMAGING AND DATA ANALYSIS CENTER

Correlation Estimation

- Recover a uniform random sample of $T_{X \bowtie Y}$ without a full join
  - Apply any correlation estimator over the sketch join

h(k)	h_u(k)	x
bac52e98	0.48	2.0
16dab449	0.34	2.0
26f79756	0.47	3.0
4da33cf5	0.34	6.0

h(k)	h_u(k)	y
bd5a7c1f	0.89	3.0
16dab449	0.34	2.5
26f79756	0.47	4.0
4da33cf5	0.34	5.0

$\bowtie$

h(k)	x	y
16dab449	2.0	2.5
26f79756	3.0	4.0
4da33cf5	6.0	5.0

$$\hat{r}_{XY} = 0.92$$

NYU TANDON SCHOOL OF ENGINEERING

VIDA VISUALIZATION IMAGING AND DATA ANALYSIS CENTER

Evaluating Join-Correlation Queries

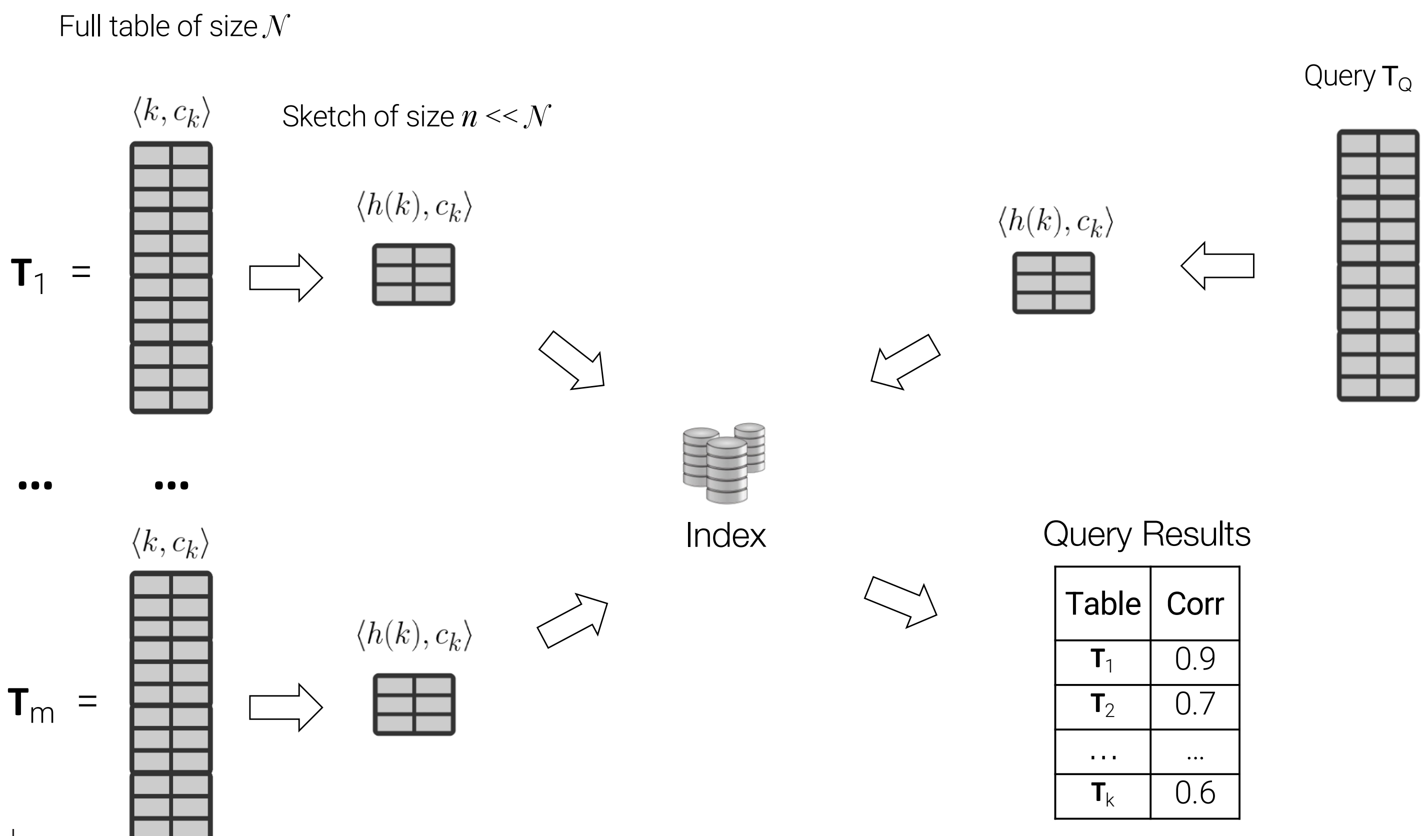
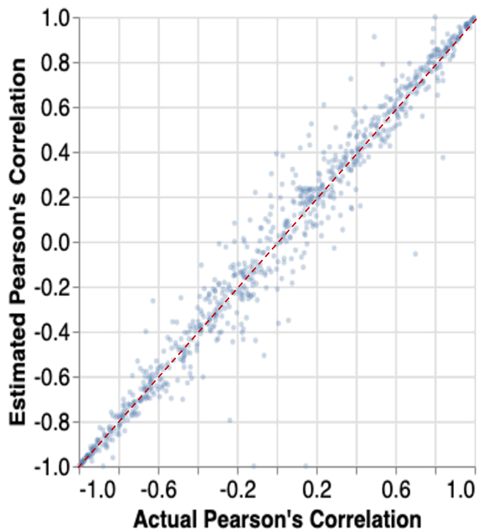


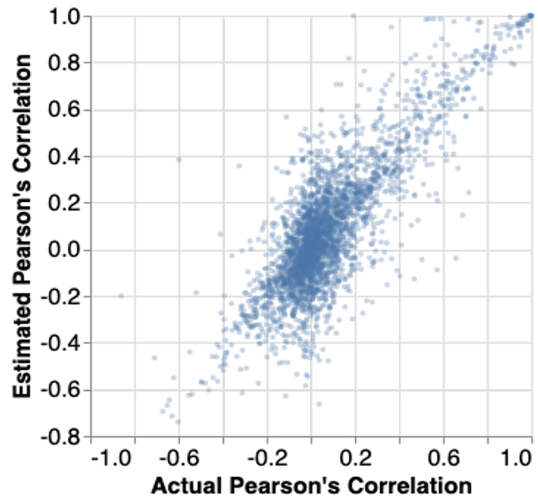
Table	Corr
T_1	0.9
T_2	0.7
...	...
T_k	0.6

[Santos et al., ACM SIGMOD 2021]

Evaluation: Estimation Accuracy



Bivariate Normal



NYC Open Data, **n ≥ 20**

It **is possible** to detect when estimates **are not good** and rank the results to avoid placing uncorrelated columns on the top (see details in [Santos et al., ACM SIGMOD 2021])

Evaluation: Estimation Performance

r_S - Spearman; r_P - Pearson; Time in ms

	Full data			**Sketch**		
percentiles	join	r_s	r_p	join	r_p	r_s
mean	42.219	8.494	0.240	0.026	0.000	0.004
std. dev.	367.696	134.357	9.314	5.618	0.042	0.279
75%	0.231	0.141	0.005	0.003	0.000	0.002
90%	7.038	0.154	0.011	0.006	0.001	0.004
99%	1360.605	29.583	0.385	0.012	0.003	0.013
99.9%	4021.838	2731.154	51.278	0.021	0.007	0.033

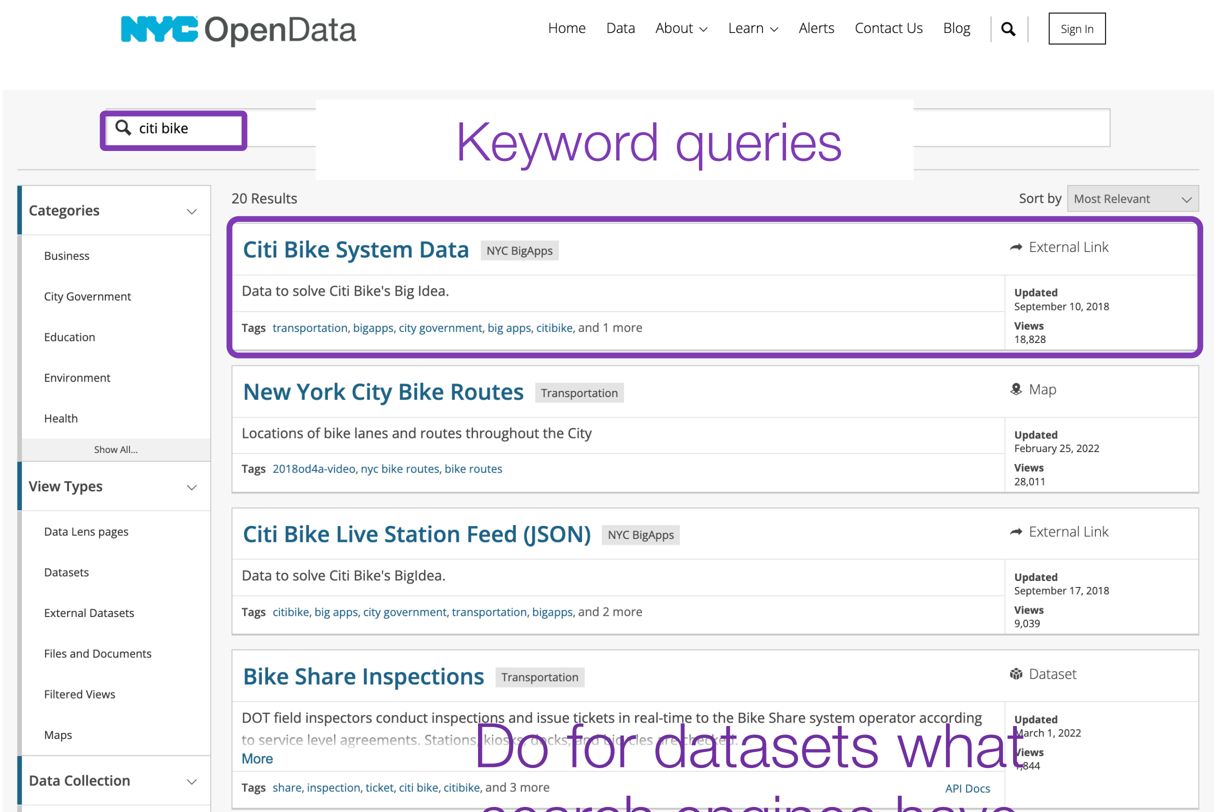
Estimates with correlation sketches take only a fraction of a millisecond.

Up to **3 orders of magnitude faster** that computing the full join!

Other Sketches

- QCR hashing [Santos et al., ICDE 2022]
  - Balance between ranking accuracy and joinability
  - Attains higher precision and recall than Correlation Sketches
- Mutual Information (MI) Sketches [Santos et al., ICDE 2024]
  - Support for numerical and categorical data
- Sketches to estimate quantities over inner products
  - Weighted sampling beats popular linear sketching methods [Bessa et al., ACM PODS 2023]
  - Efficient, linear-time sampling and accurate estimates [Daliri et al., pVLDB 2024]

Dataset Search and Discovery



Keyword queries

Textual snippets

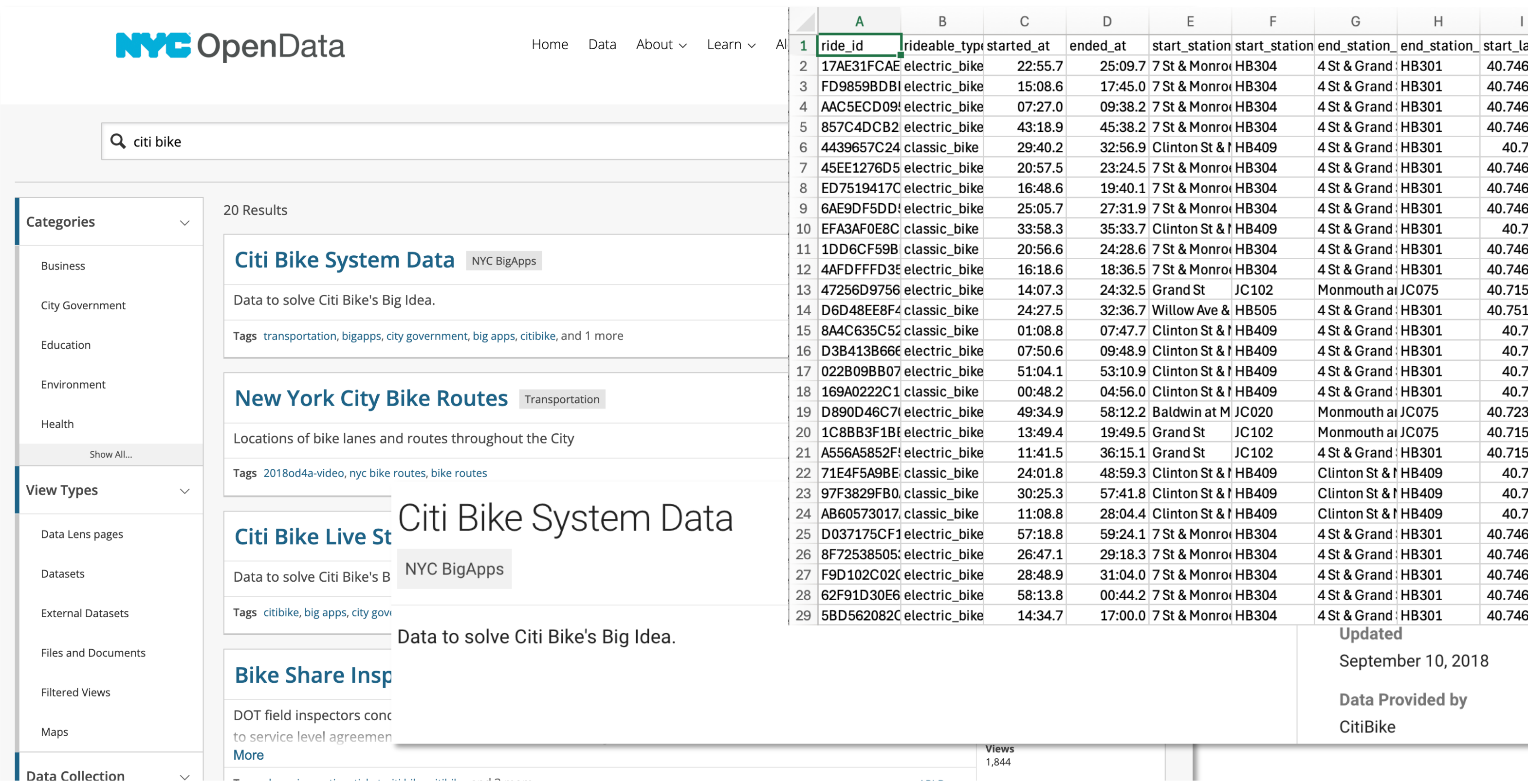
Do for datasets what search engines have done for documents

NYU TANDON SCHOOL OF ENGINEERING

VIDA VISUALIZATION IMAGING AND DATA ANALYSIS CENTER

Querying Dataset Collections

- Difficult to express information needs using keyword-based queries and to assess dataset relevance based on textual snippets

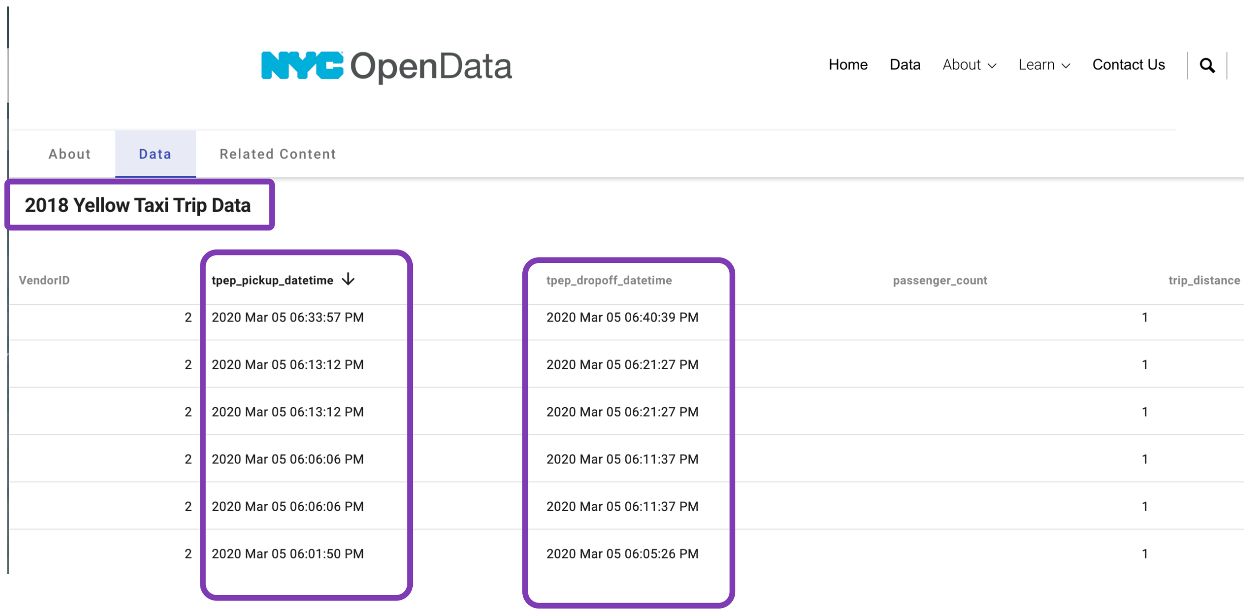


	A	B	C	D	E	F	G	H	I	J	K	L	M
1	ride_id	rideable_type	started_at	ended_at	start_station	start_station	end_station_	end_station_	start_lat	start_lng	end_lat	end_lng	member_casual
2	17AE31FCAE	electric_bike	22:55.7	25:09.7	7 St & Monro	HB304	4 St & Grand	HB301	40.7464126	-74.037977	40.742258	-74.035111	member
3	FD9859BDBI	electric_bike	15:08.6	17:45.0	7 St & Monro	HB304	4 St & Grand	HB301	40.7464126	-74.037977	40.742258	-74.035111	member
4	AAC5ECD09	electric_bike	07:27.0	09:38.2	7 St & Monro	HB304	4 St & Grand	HB301	40.7464126	-74.037977	40.742258	-74.035111	member
5	857C4DCB2	electric_bike	43:18.9	45:38.2	7 St & Monro	HB304	4 St & Grand	HB301	40.7464126	-74.037977	40.742258	-74.035111	member
6	4439657C24	classic_bike	29:40.2	32:56.9	Clinton St & I	HB409	4 St & Grand	HB301	40.73743	-74.03571	40.742258	-74.035111	member
7	45EE1276D5	electric_bike	20:57.5	23:24.5	7 St & Monro	HB304	4 St & Grand	HB301	40.7464126	-74.037977	40.742258	-74.035111	member
8	ED7519417C	electric_bike	16:48.6	19:40.1	7 St & Monro	HB304	4 St & Grand	HB301	40.7464126	-74.037977	40.742258	-74.035111	member
9	6AE9DF5DD	electric_bike	25:05.7	27:31.9	7 St & Monro	HB304	4 St & Grand	HB301	40.7464126	-74.037977	40.742258	-74.035111	member
10	EFA3AF0E8C	classic_bike	33:58.3	35:33.7	Clinton St & I	HB409	4 St & Grand	HB301	40.73743	-74.03571	40.742258	-74.035111	member
11	1DD6CF59B	classic_bike	20:56.6	24:28.6	7 St & Monro	HB304	4 St & Grand	HB301	40.7464126	-74.037977	40.742258	-74.035111	member
12	4AFDFFFD35	electric_bike	16:18.6	18:36.5	7 St & Monro	HB304	4 St & Grand	HB301	40.7464126	-74.037977	40.742258	-74.035111	member
13	47256D9756	electric_bike	14:07.3	24:32.5	Grand St	JC102	Monmouth ai	JC075	40.7151777	-74.037683	40.7256855	-74.04879	member
14	D6D48EE8F4	classic_bike	24:27.5	32:36.7	Willow Ave &	HB505	4 St & Grand	HB301	40.7518675	-74.030377	40.742258	-74.035111	member
15	8A4C635C52	classic_bike	01:08.8	07:47.7	Clinton St & I	HB409	4 St & Grand	HB301	40.73743	-74.03571	40.742258	-74.035111	casual
16	D3B413B666	electric_bike	07:50.6	09:48.9	Clinton St & I	HB409	4 St & Grand	HB301	40.73743	-74.03571	40.742258	-74.035111	member
17	022B09BB07	electric_bike	51:04.1	53:10.9	Clinton St & I	HB409	4 St & Grand	HB301	40.73743	-74.03571	40.742258	-74.035111	member
18	169A0222C1	classic_bike	00:48.2	04:56.0	Clinton St & I	HB409	4 St & Grand	HB301	40.73743	-74.03571	40.742258	-74.035111	member
19	D890D46C7	electric_bike	49:34.9	58:12.2	Baldwin at M	JC020	Monmouth ai	JC075	40.7236589	-74.064194	40.7256855	-74.04879	member
20	1C8BB3F1BE	electric_bike	13:49.4	19:49.5	Grand St	JC102	Monmouth ai	JC075	40.7151777	-74.037683	40.7256855	-74.04879	member
21	A556A5852F	electric_bike	11:41.5	36:15.1	Grand St	JC102	4 St & Grand	HB301	40.7151777	-74.037683	40.742258	-74.035111	member
22	71E4F5A9BE	classic_bike	24:01.8	48:59.3	Clinton St & I	HB409	Clinton St & I	HB409	40.73743	-74.03571	40.73743	-74.03571	casual
23	97F3829FB0	classic_bike	30:25.3	57:41.8	Clinton St & I	HB409	Clinton St & I	HB409	40.73743	-74.03571	40.73743	-74.03571	member
24	AB60573017	classic_bike	11:08.8	28:04.4	Clinton St & I	HB409	Clinton St & I	HB409	40.73743	-74.03571	40.73743	-74.03571	member
25	D037175CF1	electric_bike	57:18.8	59:24.1	7 St & Monro	HB304	4 St & Grand	HB301	40.7464126	-74.037977	40.742258	-74.035111	member
26	8F72538505	electric_bike	26:47.1	29:18.3	7 St & Monro	HB304	4 St & Grand	HB301	40.7464126	-74.037977	40.742258	-74.035111	member
27	F9D102C02	electric_bike	28:48.9	31:04.0	7 St & Monro	HB304	4 St & Grand	HB301	40.7464126	-74.037977	40.742258	-74.035111	member
28	62F91D30E6	electric_bike	58:13.8	00:44.2	7 St & Monro	HB304	4 St & Grand	HB301	40.7464126	-74.037977	40.742258	-74.035111	member
29	5BD562082C	electric_bike	14:34.7	17:00.0	7 St & Monro	HB304	4 St & Grand	HB301	40.7464126	-74.037977	40.742258	-74.035111	member

VIDA
VISUALIZATION IMAGING AND DATA ANALYSIS CENTER

Querying Dataset Collections

- Difficult to express data discovery information needs using keyword-based queries and to assess dataset relevance based on textual snippets
- Metadata is necessarily incomplete and sometimes inconsistent with the data

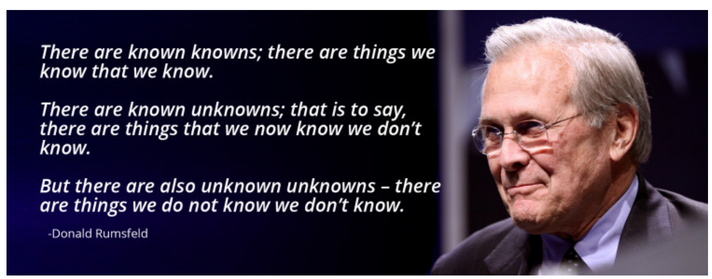


Querying Dataset Collections

- Difficult to express data discovery information needs using keyword-based queries and to assess dataset relevance based on textual snippets
- Metadata is necessarily incomplete and sometimes inconsistent with the data
- *Mismatch between users' requirements and metadata + search capabilities* [Papenmeier et al., 2021]

"complex information needs seem to collide with the capabilities of data search systems."

- Search for what you know – limited support for discovering *unknown unknowns*



NYU TANDON SCHOOL OF ENGINEERING

VIDA VISUALIZATION IMAGING AND DATA ANALYSIS CENTER

Rethinking Dataset Search and Discovery

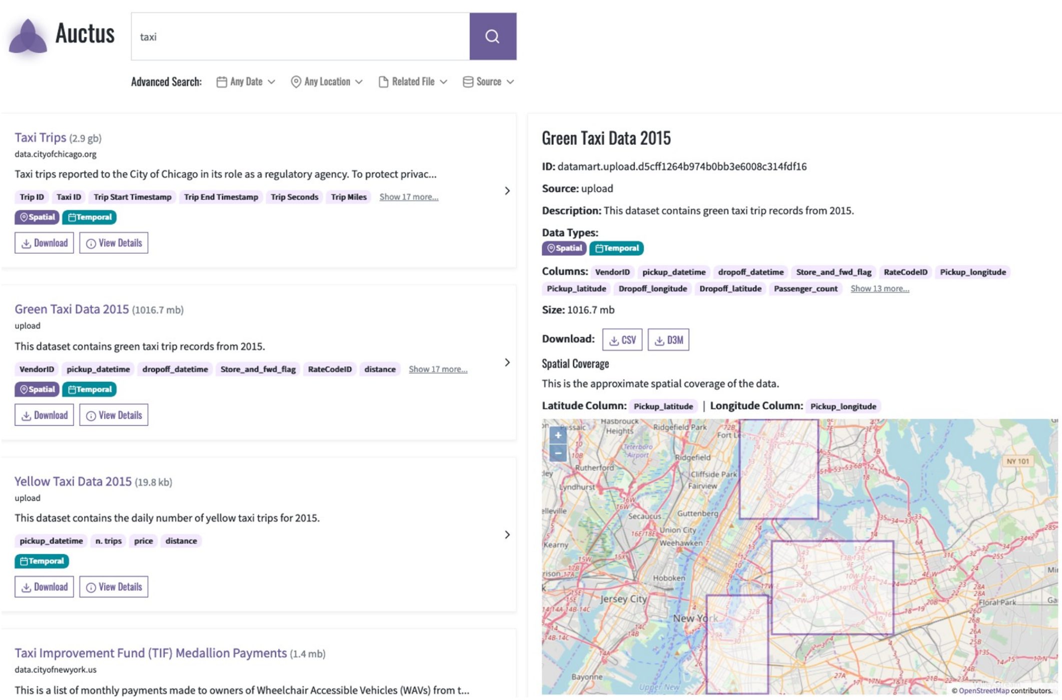
- *Dataset Relationship Queries:* **expressive queries** to search dataset collections that capture diverse information needs – *dataset as a query*
  - Find correlated variables in joinable datasets Join-Correlation – *improve ML models, test hypothesis* [Santos et al., ACM SIGMOD 2021, ICDE 2022, ICDE 2024]
  - Explain salient features in spatio-temporal data – *use data to explain data* [Chirigati et al., ACM SIGMOD 2016; Chan et al., ACM SIGMOD 2017]
  - Explain outliers in time series data – *use data to explain data* [Bessa et al., ACM TDS 2020]

Rethinking Dataset Search and Discovery

- *Profiling & Indexing:* Go beyond the metadata provided by data publishers -- leverage dataset **contents derive metadata and improve findability**
  - Rule-based type detection, e.g., categorical, numerical, spatial, temporal (https://github.com/VIDA-NYU/auctus/tree/master/lib_profiler)
  - LLM-based column type annotation: discover semantic types that capture what the data is about [Feuer et al., pVLDB 2024]
    - {BRONX, BROOKLYN, MANHATTAN, QUEENS, STATEN ISLAND} → NYC boroughs
    - {AVERNE, ASTORIA, BAYSIDE, BELLEROSE, BRIARWOOD, CORONA, ELMHURST, FAR ROCKAWAY, FLUSHING, JAMAICA, ...} → Queens neighborhoods

Rethinking Dataset Search and Discovery

- *Interaction, informative snippets and result presentation:* facilitate exploration and identification of relevant data [Castelo et al., PVLDB 2021]



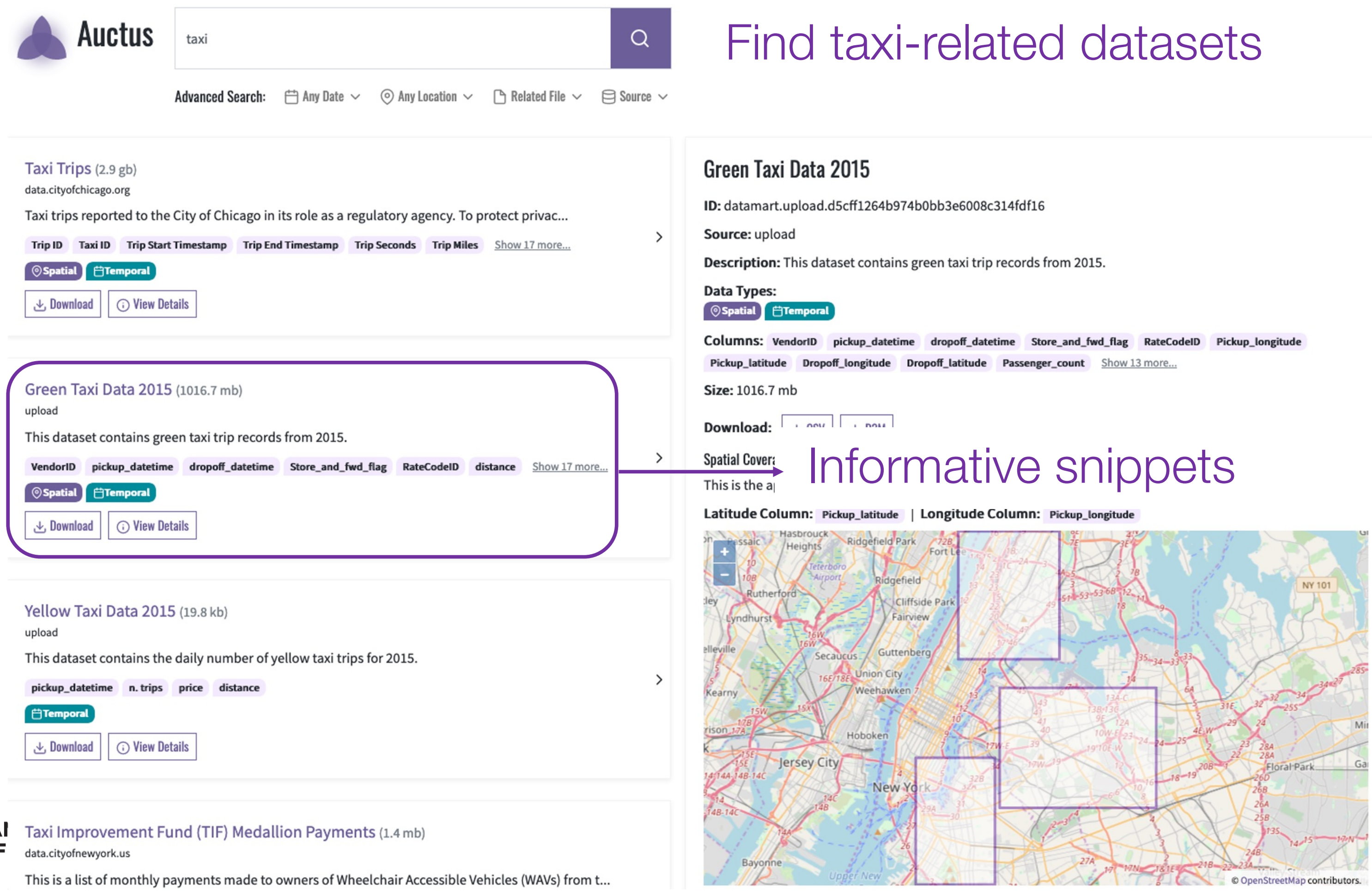
Auctus Dataset Search Engine

https://auctus.vida-nyu.org/

NYU TANDON SCHOOL OF ENGINEERING

VISUALIZATION IMAGING AND DATA ANALYSIS CENTER

Keyword-Based Search



Find taxi-related datasets

Informative snippets

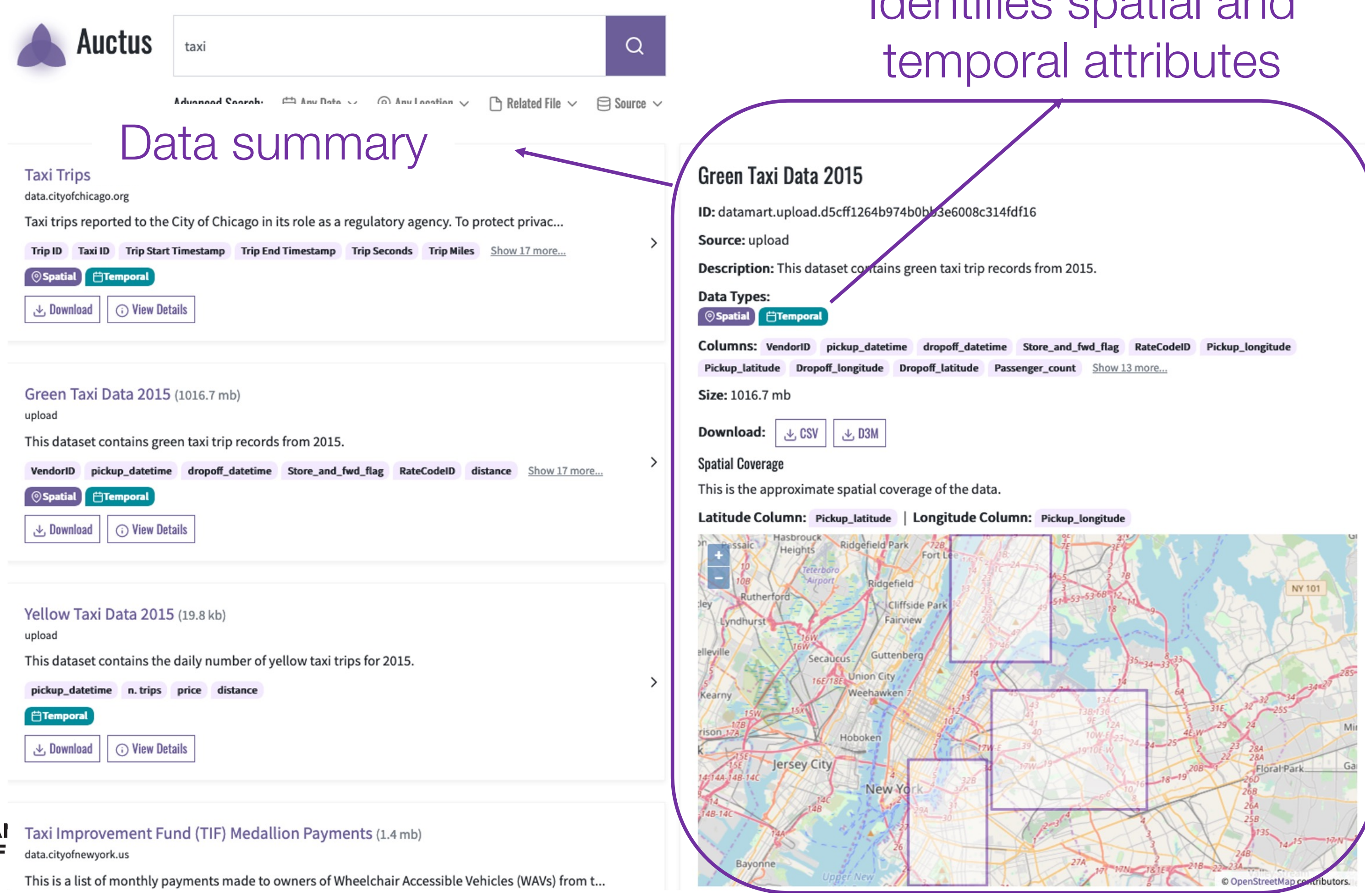
NYU

VISUALIZATION IMAGING AND DATA ANALYSIS CENTER

Visualizing: Automatically Inferred Metadata

Identifies spatial and temporal attributes

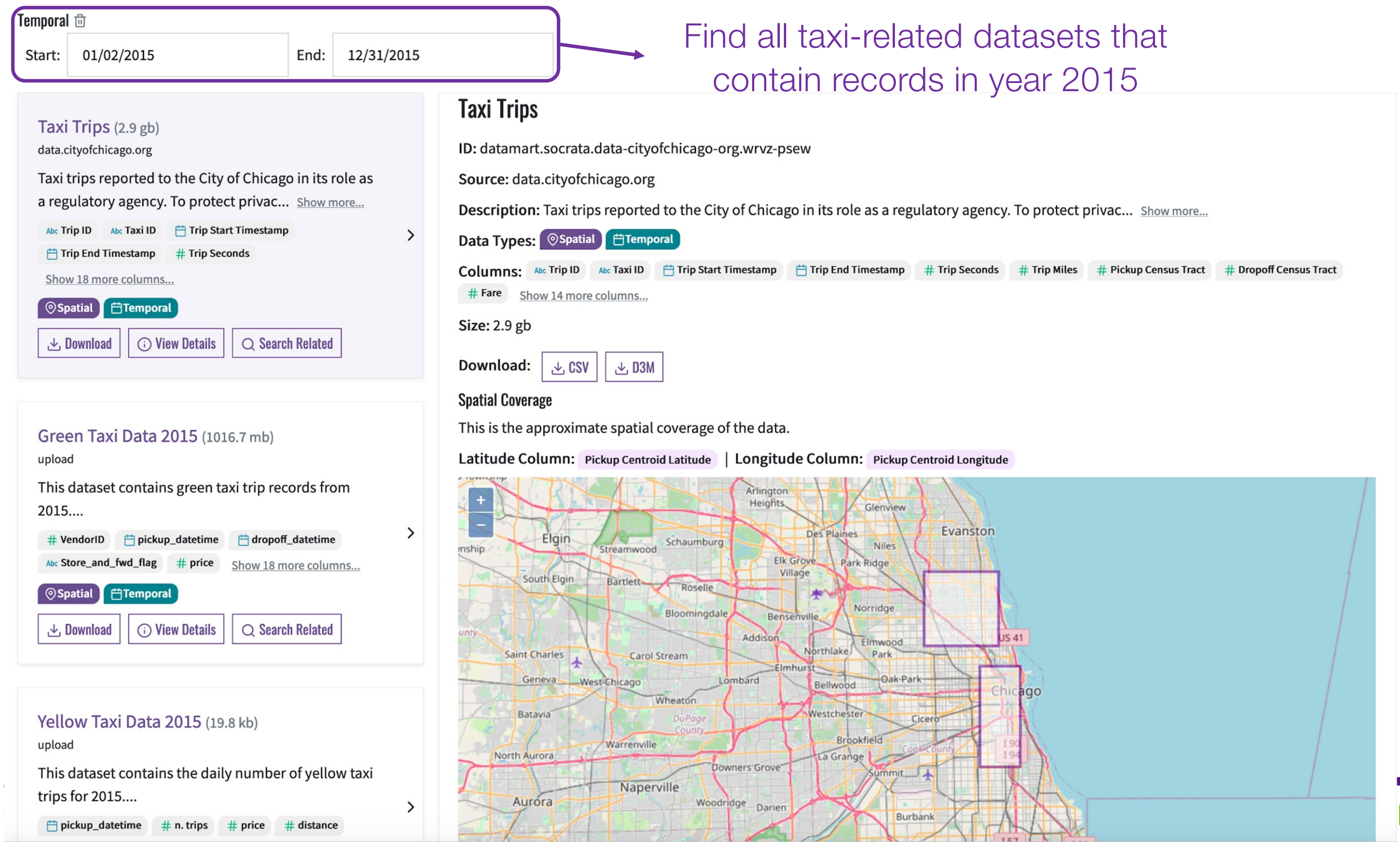
Data summary



NYU

VISUALIZATION IMAGING AND DATA ANALYSIS CENTER

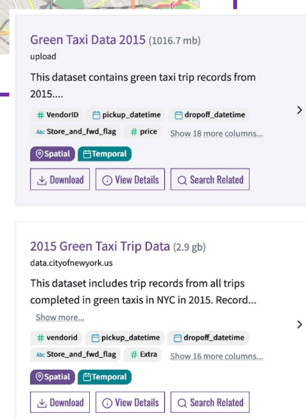
Filtering by Time

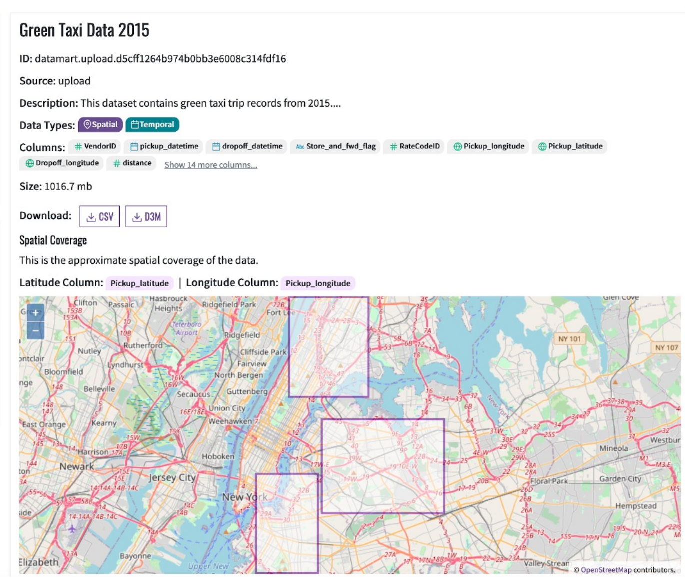


Find all taxi-related datasets that contain records in year 2015

Filtering by Space

Find all taxi-related datasets that contain records in year 2015 and that cover the NYC area





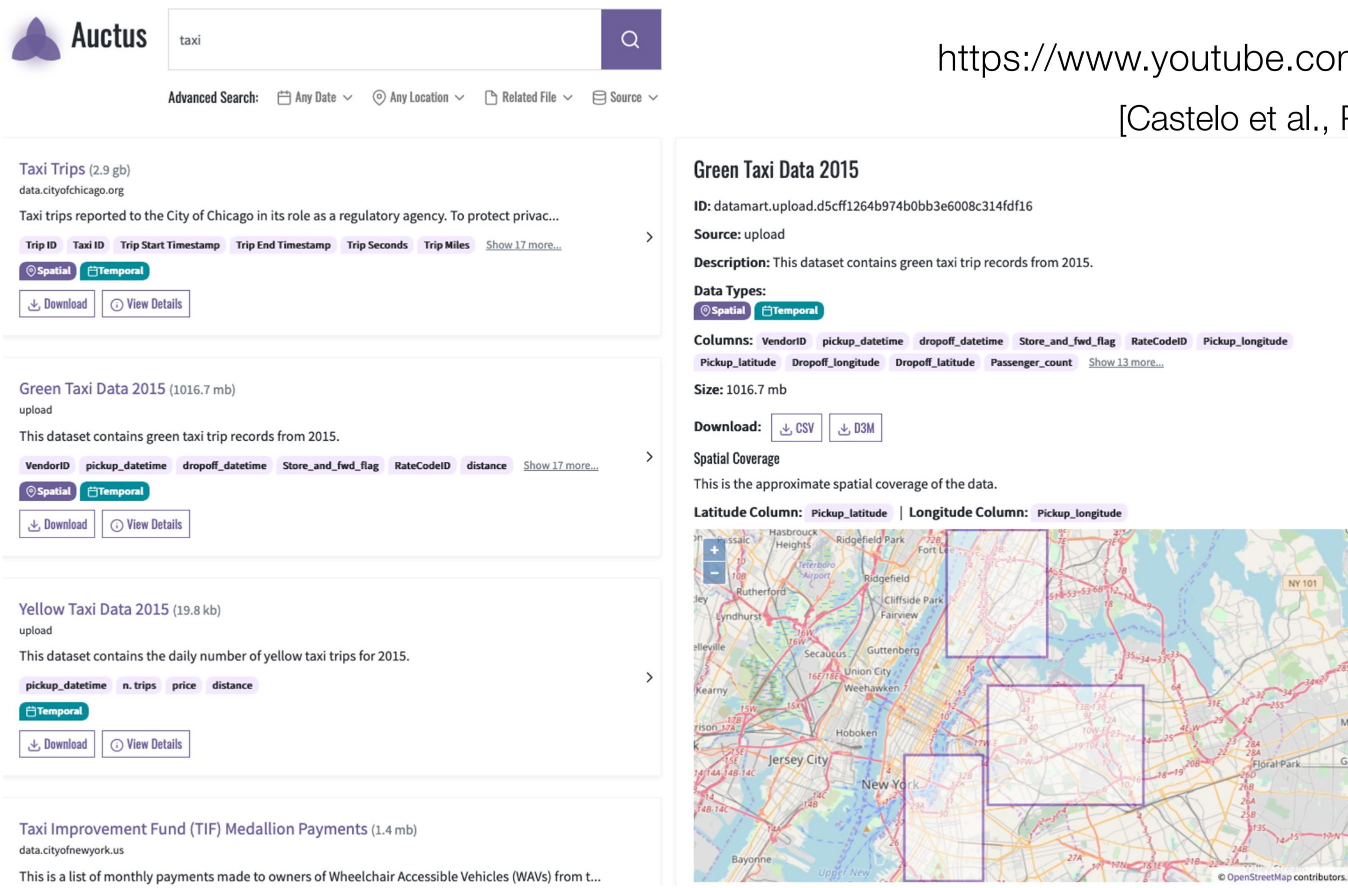
NYU | TANDON SCHOOL OF ENGINEERING

VIDA VISUALIZATION IMAGING AND DATA ANALYSIS CENTER

The Auctus Dataset Search Engine

https://www.youtube.com/watch?v=lZQbh3ctq6Q

[Castelo et al., PVLDB 2021]



NYU TANDON SCHOOL OF ENGINEERING

DARPA

VIDA VISUALIZATION IMAGING AND DATA ANALYSIS CENTER

Conclusions

- Artificial Intelligence (AI) is reshaping data-driven exploration – it is augmenting, not replacing users: need the user in the loop
- AutoML is democratizing ML *to a certain extent*
  - Automation is not enough – need explainability and trust, including the ability to debug pipelines [Lourenco et al., VLDBJ 2022]
  - Clearly helps data scientists and computer-literate users
  - Practicing ML is still hard for domain experts

Will LLMs render AutoML systems obsolete? Or *improve* them?

LLMs also need explainability and trust

Conclusions (cont.)

- There is a huge untapped value in open data, internal repositories and data lakes – it is hard to find *relevant* data
  - Rethink the design and implementation of dataset search engines
  - Data discovery by uncovering data relationships – dataset queries
- Finding data is only the first step…need to assess quality, curate, and integrate data
  - LLMs can help with data wrangling [Feuer et al., pVLDB 2024, Kayali et al., pVLDB 2024, Narayan et al., pVLDB 2022, and **many** others]

Acknowledgments

Merci

Gracias

Ευχαριστω

با تشکر

謝謝

고맙습니다

Thank you

Obrigada

благодаря

Kiitos

धन्यवाद

Tack

Danke

Bedankt

NYU TANDON SCHOOL OF ENGINEERING

VIDA VISUALIZATION IMAGING AND DATA ANALYSIS CENTER