Data Placement Optimization for ATLAS in a Multi-Tiered Storage System within a Data Center

Qiulan Huang, James Leonardi, Carlos Deleon,Vincent Garonne, Shinjae Yoo, Carlos Gamboa

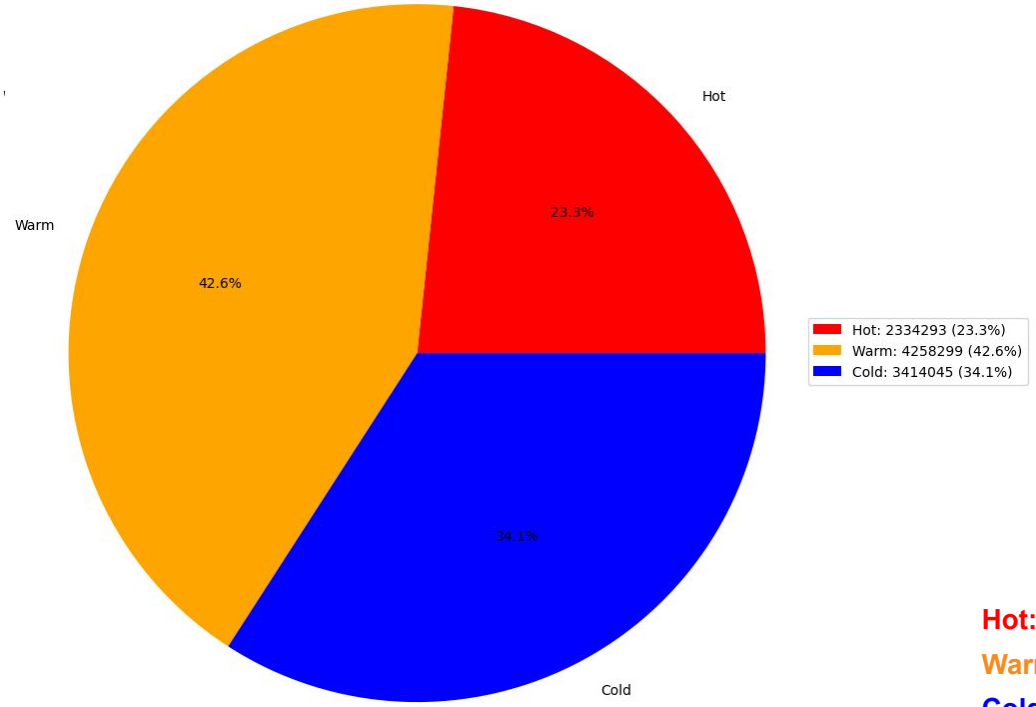
Brookhaven National Laboratory

@BrookhavenLab

CHEP 2024, Krakow, Poland - Oct 19 - Mar 25, 2024

Data Temperature（Take BNL ATLAS data for example）

Aug 25, 2023-Aug 23, 2024, ~10 million files



Hot: Last access in the last month

Warm: Last access in the last 6 months

Cold: Last access between 6 months and one year

Brookhaven National Laboratory

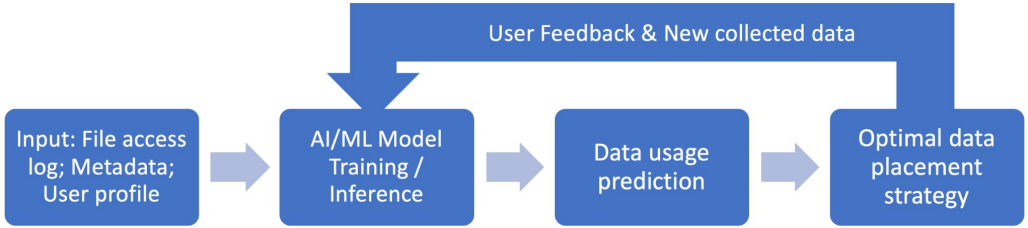
AI/ML For Storage Optimization

Motivation

- In the current tiered storage "class" system at the Data Center
  - Unused data is stored on faster IO disk storage
  - Explore whether fast IO storage can be used more effectively

Goals

- Design an efficient monitoring platform to collect the relevant information from various distributed data sources
- Develop an optimal data management system for the data center to maximize usable space while minimizing access latency, within budget, hardware, and compliance constraints
  - Heavy use of storage, metadata and data popularity information
  - Develop a precise AI/ML prediction model to possibly forecast the future usage of the data
  - **Orchestration of data for optimal movement and placement**



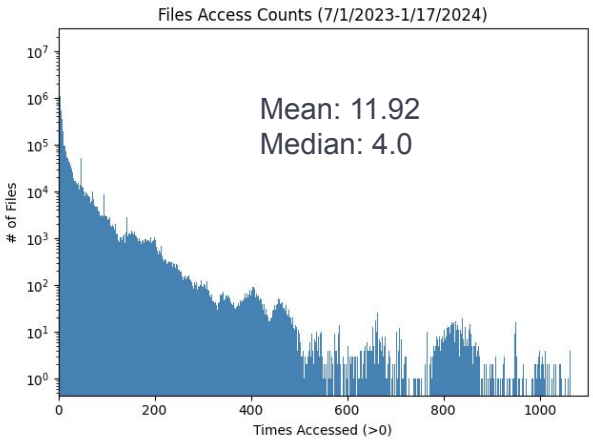
Data collection and preprocessing

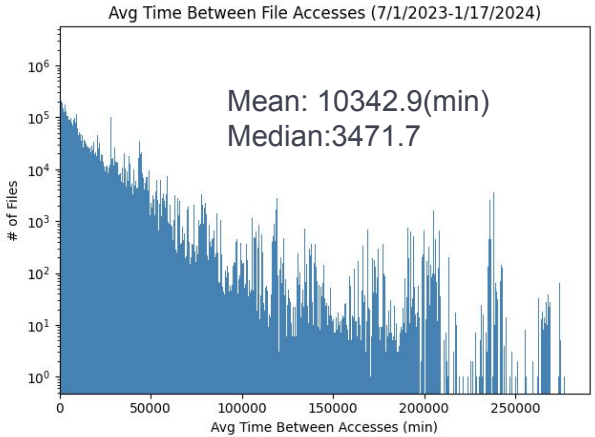
- **~11 TB** data is collected from various data sources from disk storage system like dCache billing logs, domain logs, etc
  - ~10GB in average per day, 5~8 million events per day
- Define and generate the tabular data or comma-separated values (CSV) file format for data training and facilitates finding patterns between files
  - pnfsid
  - Access Count
  - Access Timestamps
  - Rucio Scope (mc15_13TeV)
  - Task ID
  - Datatype (DAOD, EVNT, HIST, etc.)
  - Avg Time Between Accesses
  - Action(create, transfer, delete,)
  - User ID
  - …

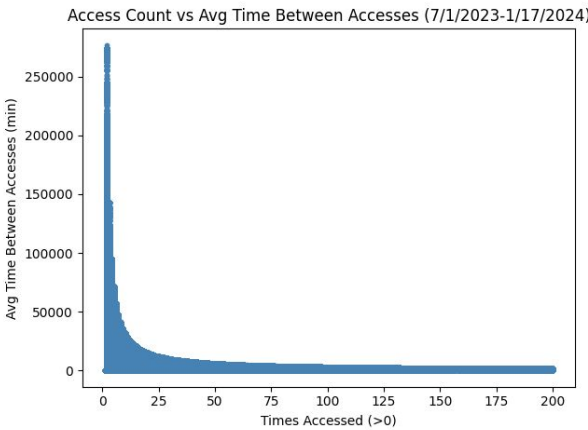
File ID	path	taskid	datatype	scope	First_Access	Last_Access	…
file_1							
file_2							
…							

```
pnfsid|path|taskid|datatype|scope|accesscount|clientips|protocols|actions|firstaccess|accesstimes|lastaccess|mintimebetween|avgtimebetween|maxtimebetween|errorcodes
0000A3EECFE022224142A68A0037FE3A446D|/pnfs/usatlas.bnl.gov/BNLT0D1/rucio/mc23_13p6TeV/d6/ad/DAOD_PHYSLITE.35040159._000250.pool.root.1|35040159|DAOD_PHYSLITE|mc23_13p6TeV|1|{'130.199.206.137'}|{'Xrd
otd-5.0'}|{'request'}|2023-11-01 00:00:02.540000-0400|{'2023-11-01 00:00:02.540000-0400'}|2023-11-01 00:00:02.540000-0400|0|0|0|{'0'}
00008583BBF8DD8A4B0787679565564E2794|/pnfs/usatlas.bnl.gov/BNLT0D1/rucio/mc23_13p6TeV/1e/d6/DAOD_PHYSLITE.35040159._000342.pool.root.1|35040159|DAOD_PHYSLITE|mc23_13p6TeV|1|{'130.199.206.149'}|{'Xrd
otd-5.0'}|{'request'}|2023-11-01 00:00:05.428000-0400|{'2023-11-01 00:00:05.428000-0400'}|2023-11-01 00:00:05.428000-0400|0|0|0|{'0'}
000058B7CD9318E44138857679E39F0E5B17|/pnfs/usatlas.bnl.gov/BNLT0D1/rucio/mc23_13p6TeV/a4/60/DAOD_PHYSLITE.35040159._000330.pool.root.1|35040159|DAOD_PHYSLITE|mc23_13p6TeV|1|{'130.199.156.199'}|{'Xrd
otd-5.0'}|{'request'}|2023-11-01 00:00:06.400000-0400|{'2023-11-01 00:00:06.400000-0400'}|2023-11-01 00:00:06.400000-0400|0|0|0|{'0'}
000058BC8CE6F325496B982EF0ABF2B2AF05|/pnfs/usatlas.bnl.gov/BNLT0D1/rucio/mc23_13p6TeV/7d/9c/DAOD_PHYSLITE.35040159._000253.pool.root.1|35040159|DAOD_PHYSLITE|mc23_13p6TeV|1|{'130.199.159.140'}|{'Xrd
otd-5.0'}|{'request'}|2023-11-01 00:00:06.777000-0400|{'2023-11-01 00:00:06.777000-0400'}|2023-11-01 00:00:06.777000-0400|0|0|0|{'0'}
0000223C108F5ED14EB59CAA13263B97E30F|/pnfs/usatlas.bnl.gov/BNLT0D1/rucio/mc20_13TeV/01/97/AOD.35261114._000644.pool.root.1|35261114|AOD|mc20_13TeV|2|{'130.199.206.204'}|{'Xrootd-5.0'}|{'request'}|20
23-11-01 00:00:07.714000-0400|{'2023-11-01 00:00:07.714000-0400', '2023-11-01 00:00:07.757000-0400'}|2023-11-01 00:00:07.757000-0400|0.043|0.043|0.043|{'0'}
```

Data Analysis- Access Distribution



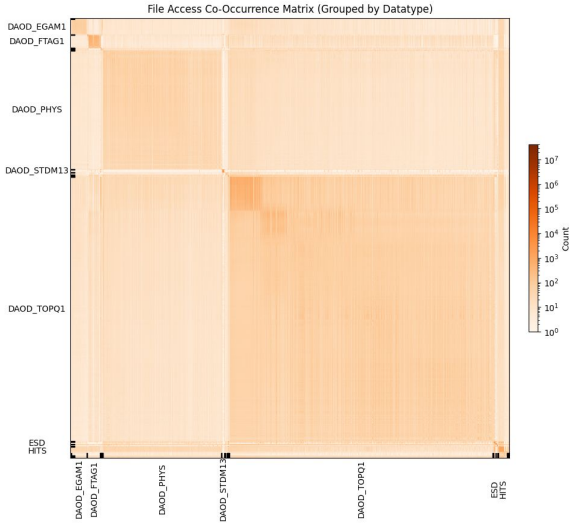


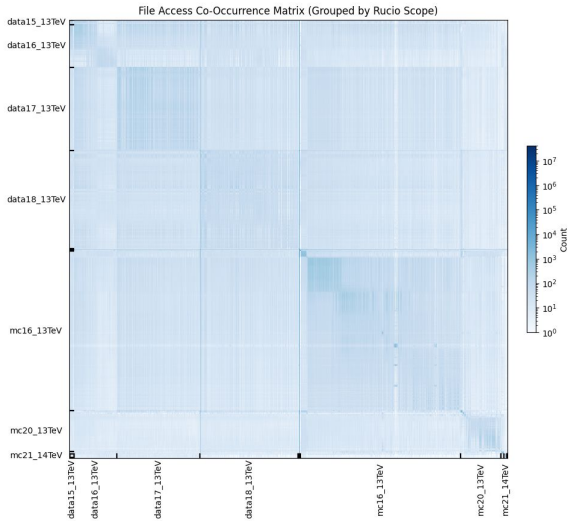


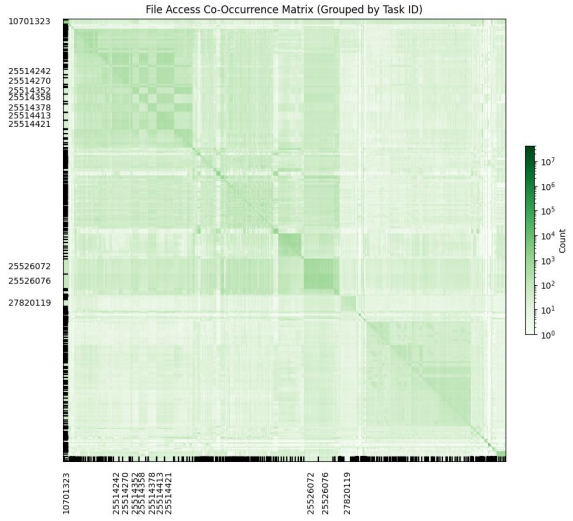
- Access count tends to taper off (some exceptions)
- As files are accessed more, time between accesses tends to decrease
- Rightmost plot trimmed to show patterns

The Data Co-occurrence Matrix

- ➔ Group by any desired attribute: Task ID, Rucio Scope, Datatype, etc.
- ➔ Patterns appear along diagonal
- ➔ Denote highly correlated groupings





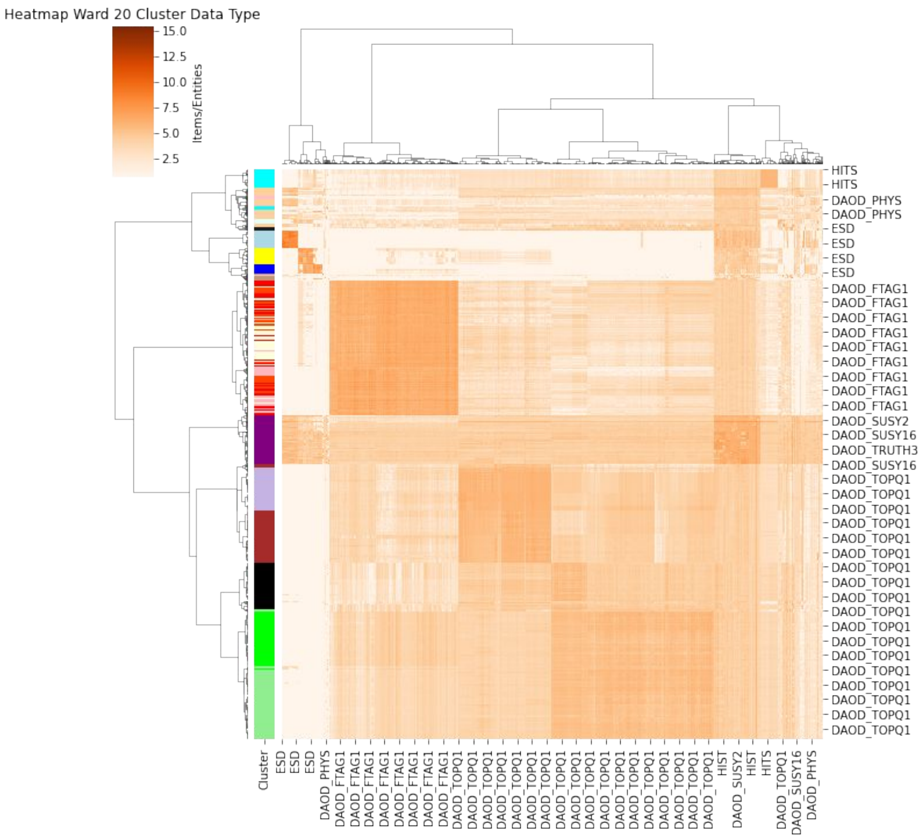


- Visualize how files are accessed with each other based on some attributes
- Strong patterns mean the attributes can be used for prediction/training
- Focus on highly-accessed files for analysis (150+ access times, 90K files)
  - The matrix size reduce from 23 million×23 million to 90K×90K

Brookhaven National Laboratory

Data Analysis- Clustering

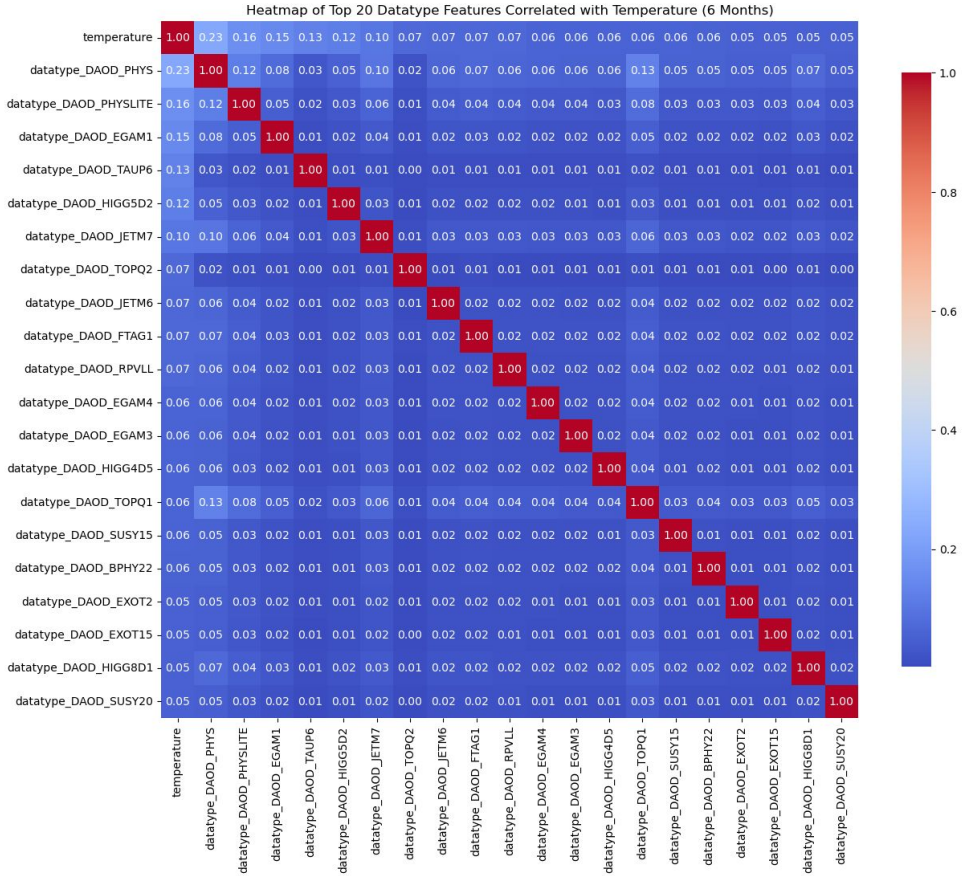
- Perform unsupervised learning
- Explore the patterns that help to differentiate the data
- A clear pattern in data type shown in the matrix correlation as well as the dendrogram hierarchical clustering and the K-means clustering
- All 3 clustering methods show a pattern is connected to the datatype feature



Brookhaven National Laboratory

Feature correlation matrices

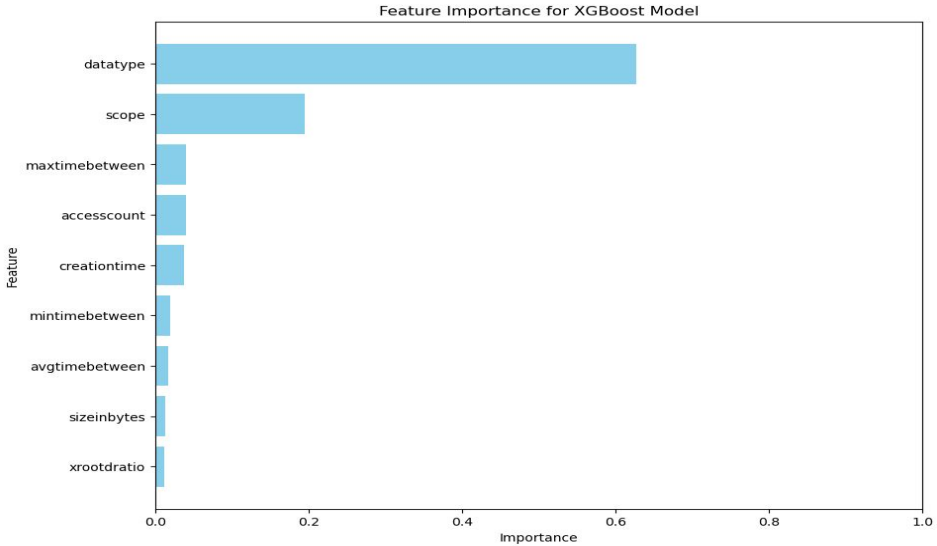
- Further to uncover the hidden patterns of data type feature, we use heatmap to show the correlation between different data types in a data sample
  - Color intensity represents the strength of the correlation with temperature as one of the key features
  - The warmer color indicate a stronger correlation with temperature



Data Training

- **Data samples**: 1 year data (~10 million files)
- **Features**:hold patterns that were shown in previous slides

  ['datatype', 'Rucio scope', 'dataset','accesscount', 'creationtime','avgtimebetween',.....]
- **Algorithms for training**:XGBoost Model using Optuna
- **Feature importance:**The features we used to train our model all impact the model differently. Some of our features impact the model more than others. The percentage of each feature tells us how much of an impact it is to the decision tree when determining the classification



Brookhaven National Laboratory

Prediction Model and Results(1)

Input/Output:

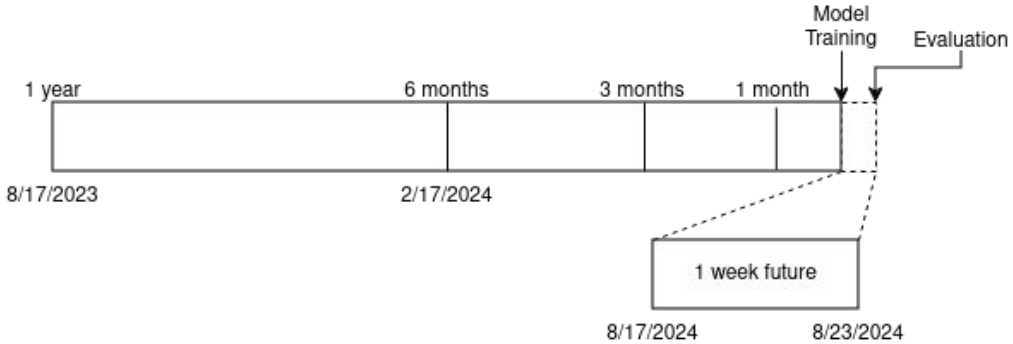
- Input of the model: preprocessed data(CSV file), introducing slide window technique to generate the input data in a fixed time window
- Output of the model: hot/cold classification

Model Training:

- Using XGBoost Model using Optuna
- Labeled data temperature based on the last accessed file
    - files accessed within the past week as "hot", while other files not accessed are labeled as "cold"

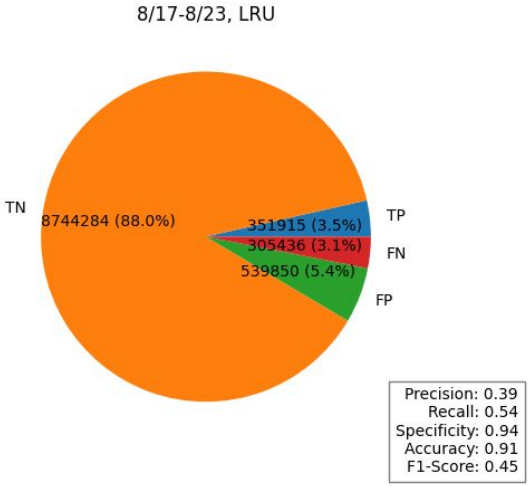
Performance Evaluation:

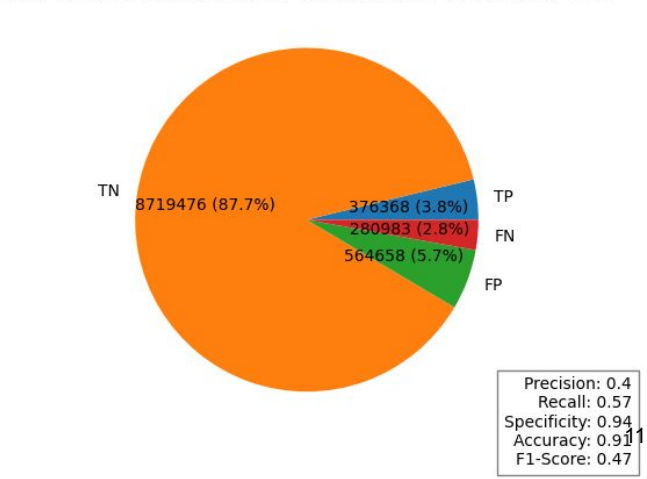
- The model's performance was evaluated on the various testing sets(spanning 1 month, 3 months, 6 months and 1 year) and validated it with the following week
- Assess its predictive accuracy, precision, specificity, recall and F1 score



Prediction Model and Results(2)

- The optimal accuracy of prediction is up to **92%** and the best F1-Score is **0.47**
- Performance improves with increasing amounts of training data
- The model slightly outperforms the LRU(Least Recently Used) policy
- In ATLAS data management strategies, data transfer often occurs at the dataset level.
    - The model provides Hot and Cold data predictions in **dataset level**

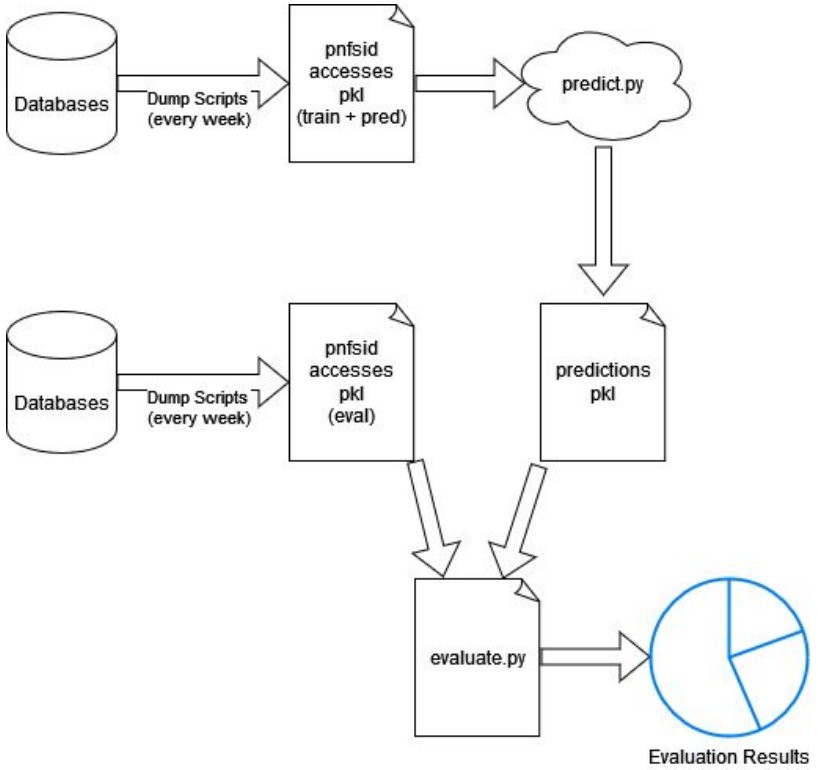




Prediction results				
	1 month	3 months	6 months	1 year
Accuracy	0.59	0.78	0.90	0.92
Precision	0.13	0.18	0.36	0.41
Recall	0.94	0.67	0.72	0.54
F1-Score	0.23	0.29	0.48	0.47

Prototype deployment

- Develop a pipeline workflow to integrate all the modules including data collection and analysis, model training, data prediction and evaluation
- Set the pipeline to run as a cron job every Saturday
- The model will generate a dataset list tagged as “hot” or “cold” (**Map file to dataset**)for the upcoming week
- At the end of each week, assess performance of the model with the actual accessed data during that week
- Data policy engine will decide the data movement according to the tagged dataset list



Summary

- The exploratory data analysis provides useful patterns for data training
- The prediction model performs well, with optimal accuracy 92% and the optimal F1-Score 0.47
- The model provides dataset level prediction for ATLAS
- The data policy engine optimizes the data placement based on the predicted data temperature
- However, deploying this model in production has many challenges, like fluctuating accuracy of the predictions at different scales. Additional testing and validation work are needed.

Thank you!