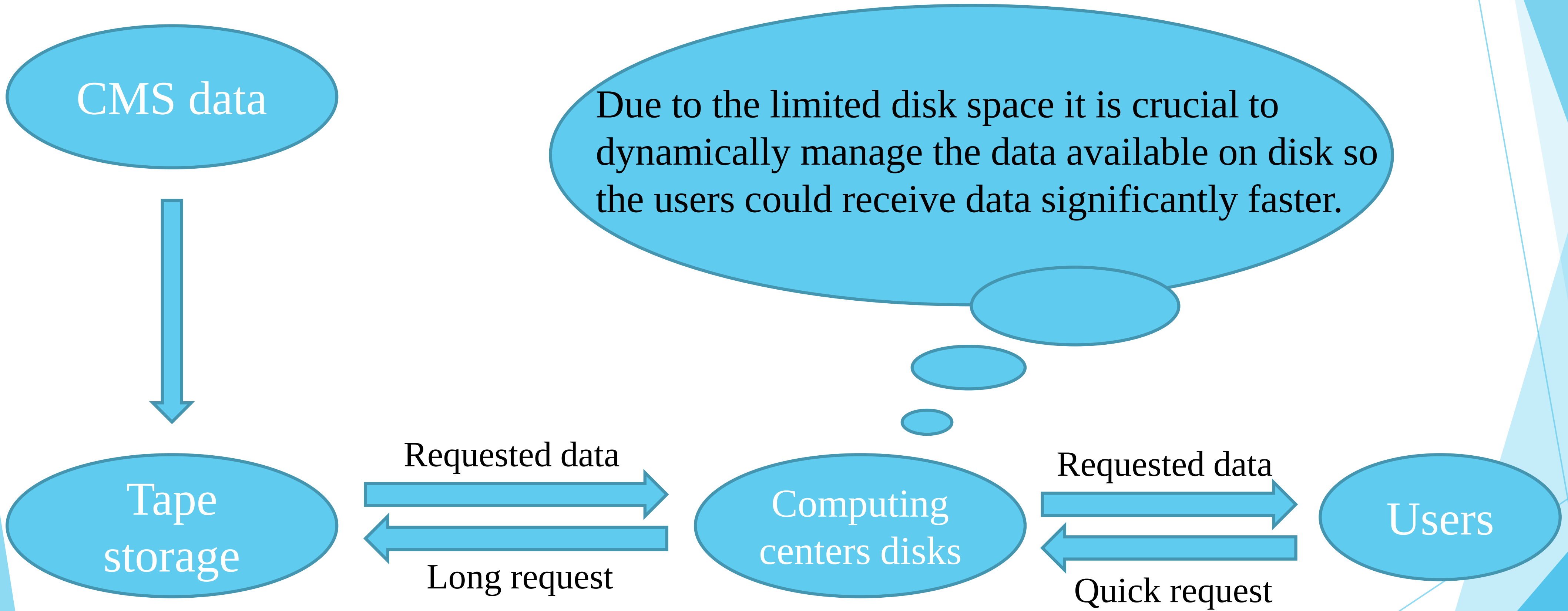
Predict CMS data popularity to improve its availability for physics analysis

Andrii Len
Taras Shevchenko National University of Kyiv

IRIS-HEP Summer Fellowship
Mentors: Dmytro Kovalskyi, Rahul Chauhan, Hasan Ozturk (MIT, CERN)

July 19, 2023

CMS data management



Due to the limited disk space it is crucial to dynamically manage the data available on disk so the users could receive data significantly faster.

CMS Remote Analysis Builder (CRAB)

- CRAB is a utility to submit CMSSW jobs to distributed computing resources.
- CRAB allows general users to access CMS data and Monte-Carlo (MC) and exploit the CPU and storage resources over there.
- CRAB saves history of tasks created by users. We will use this information.

Apache Spark on SWAN

- Apache Spark is an open-source cluster-computing framework, built around speed, ease of use, and streaming analytics.
- CRAB data could be accessed through spark cluster and be stored in the Hadoop Distributed File System (HDFS).
- CERN General Purpose (Analytix) cluster is connected via CERN SWAN (Service for Web based Analysis).

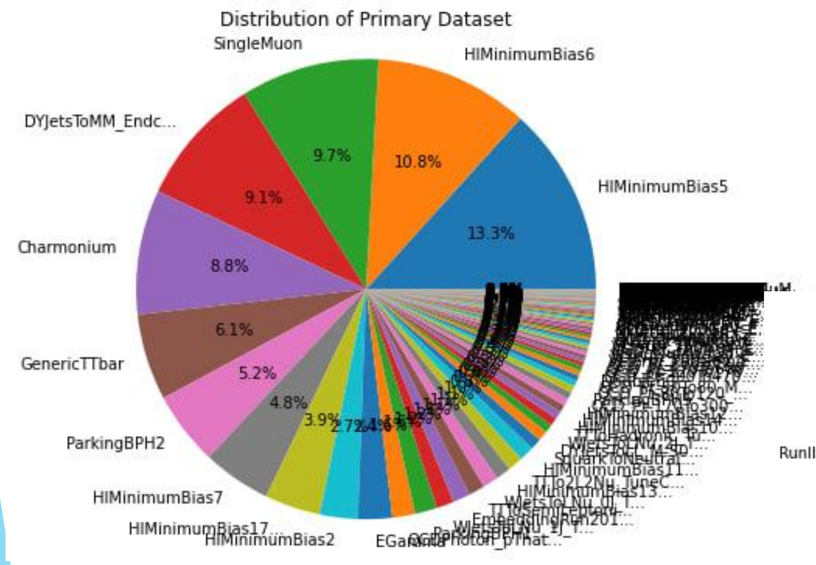
Data popularity

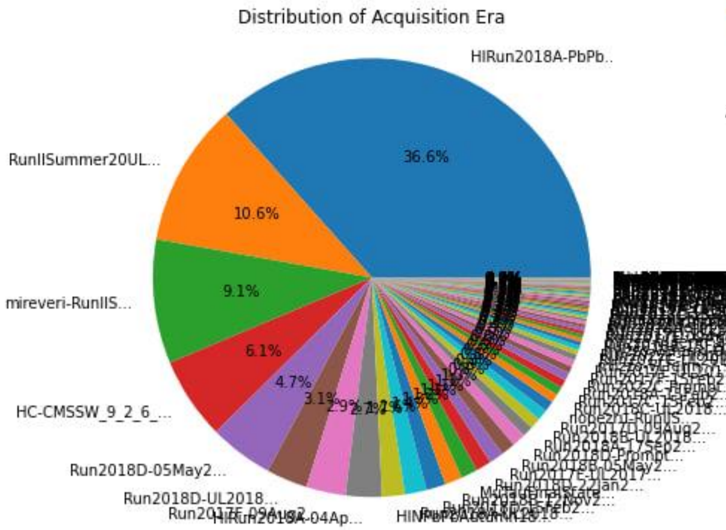
- There are various datasets but all of them have general naming rules:

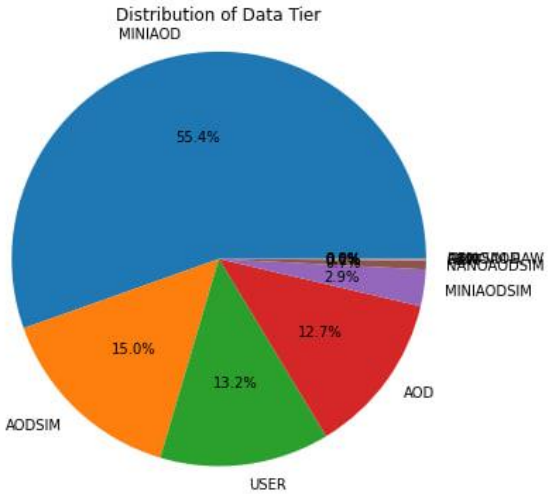
/PrimaryDataset/AcquisitionEra-ProcessingVersion/DataTier

(for example /Tau/Run2016E-HIPM_UL2016_MiniAODv2-v1/MINIAOD)

- We can extract useful information about each dataset to understand which categories are more popular:



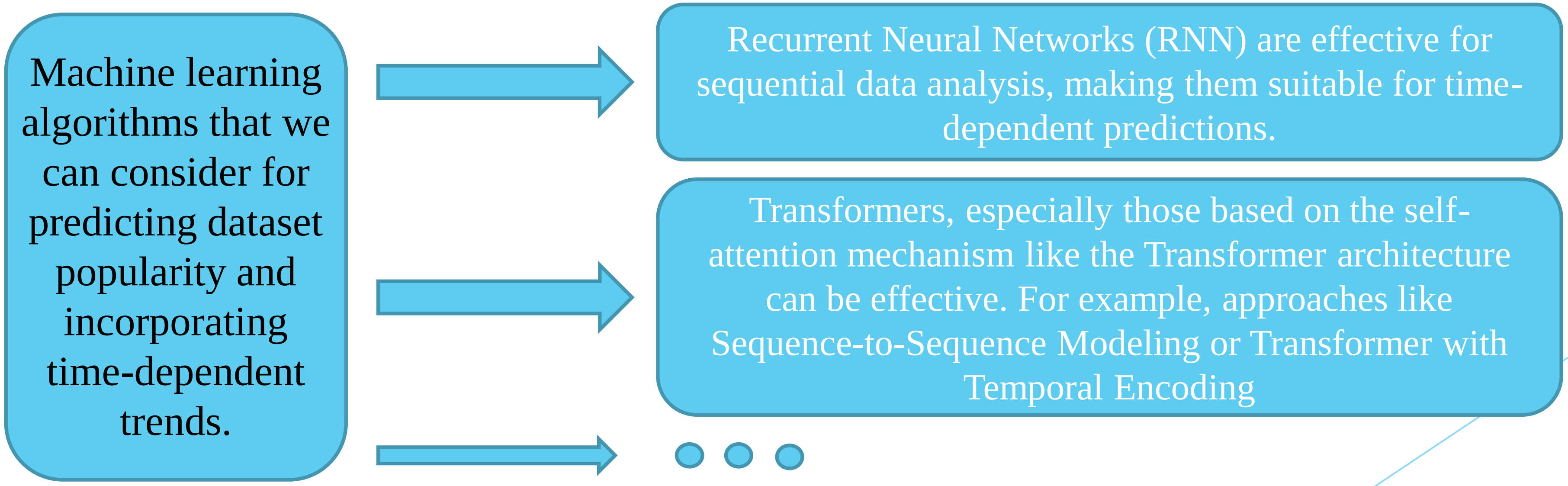




Predict data popularity based on it’s historical usage patterns

- Each task on CRAB corresponds with time stamp of it’s creation.
- Use timings and frequency of tasks to understand trends.

Machine Learning for predicting



Machine learning algorithms that we can consider for predicting dataset popularity and incorporating time-dependent trends.

Recurrent Neural Networks (RNN) are effective for sequential data analysis, making them suitable for time-dependent predictions.

Transformers, especially those based on the self-attention mechanism like the Transformer architecture can be effective. For example, approaches like Sequence-to-Sequence Modeling or Transformer with Temporal Encoding

A typical question that we want to answer:

Will a dataset be accessed in the next month?

Plan of the project

- Collection of data usage data
- Feature Engineering
- Trying various Machine Learning models
- Model Evaluation
- Integration and Deployment

Thank you for your attention!!!