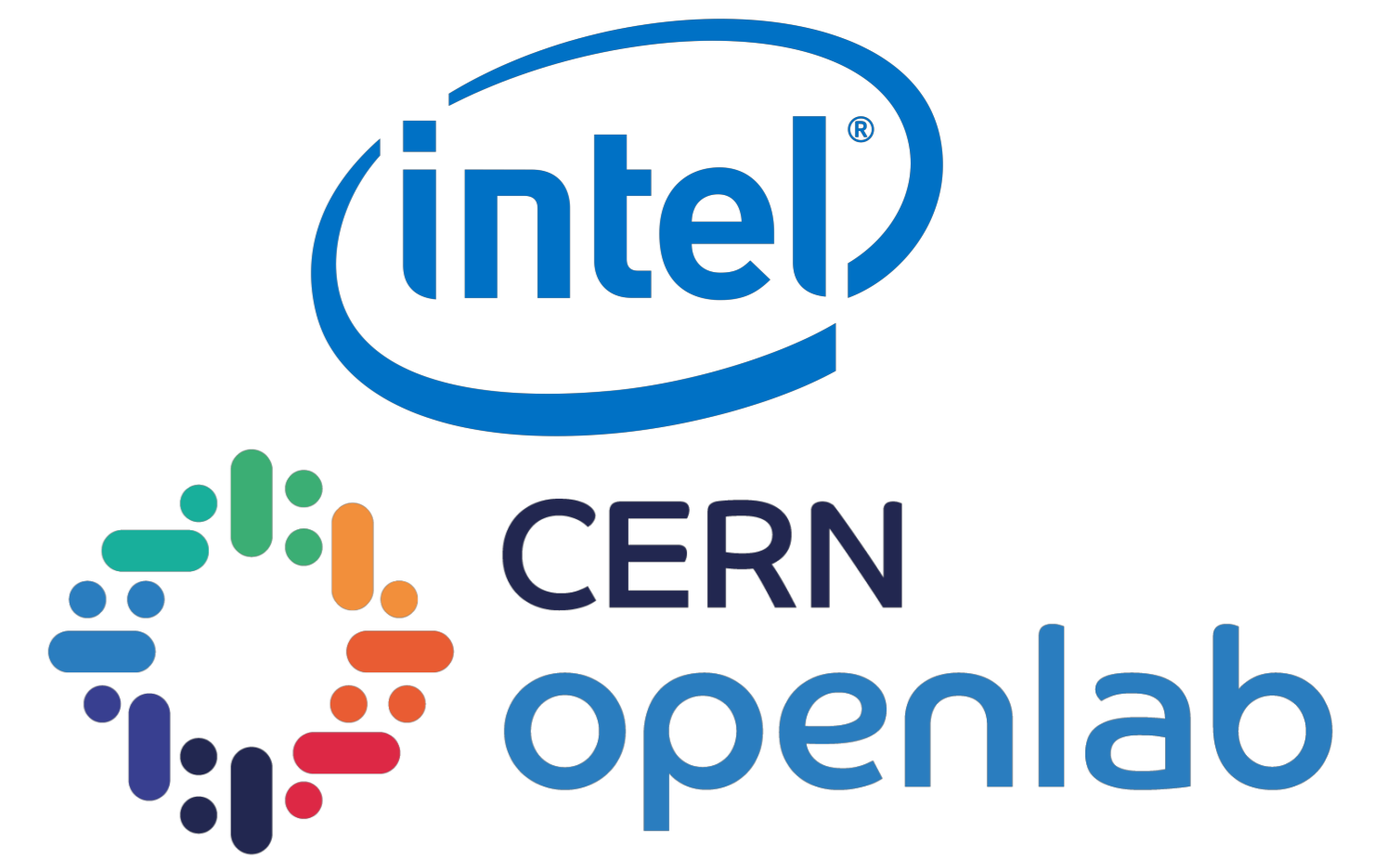


Machine Learning Pipelines with Apache Spark and Intel BigDL

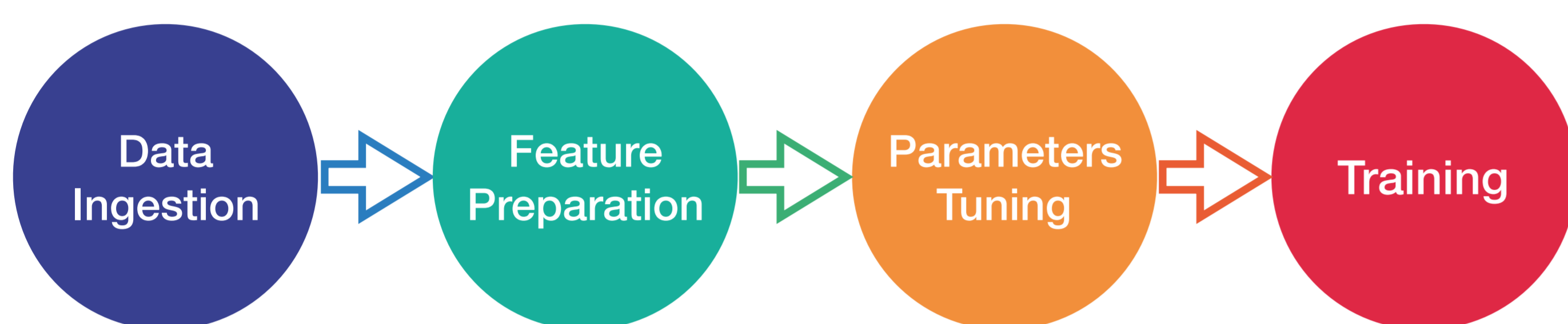


IT Information Technology Department

M. Migliorini^{1,2}, V. Khristenko¹
M. Pierini¹, E. Motesnitsalis¹, L. Canali¹, M. Girone¹
¹CERN, Geneva, Switzerland; ²University of Padova, Padova, Italy

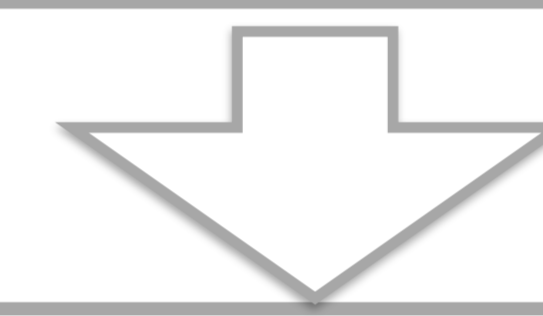
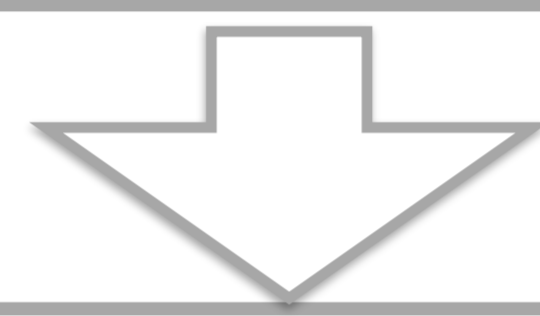
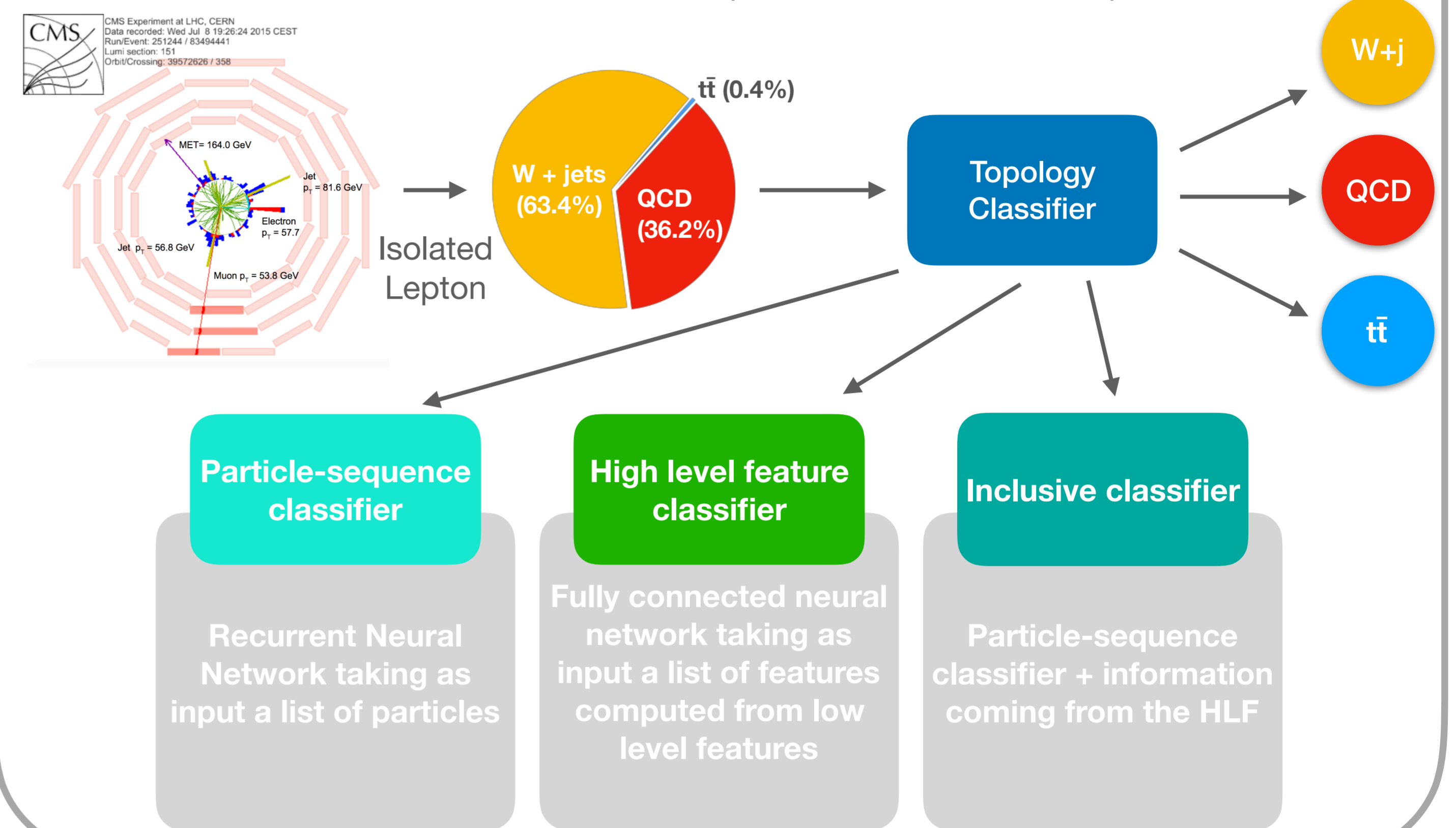
End-to-End ML Pipeline

- The goal of this work is to produce a demonstrator of an end-to-end **Machine Learning pipeline** using **Apache Spark**
- Investigate and **develop solutions** integrating:
 - Data Engineering/Big Data tools
 - Machine Learning tools
 - Data analytics platform
- Use **Industry standard tools**:
 - Well known and widely adopted
 - Open the HEP field to a larger community
- The **Pipeline** is composed by the following stages:



HEP use case

- The ability to **classify events** is of fundamental importance and **Deep Learning** proved to be able to outperform other ML methods
- See paper: "Topology classification with deep learning to improve real time event selection at LHC" (arXiv:1807.00083v2)



Data Ingestion



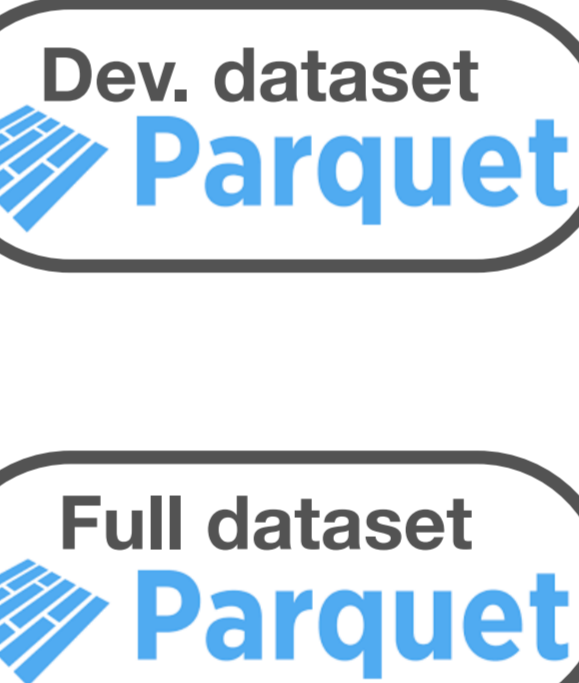
EOS storage
Input:

- 10 TB of ROOT files
- 50M events

- Access physics data stored in EOS using **Hadoop-XRootD Connector**
- Read ROOT files into a **Spark DF** using **Spark-ROOT** reader

Feature Preparation

- Filter events**: require the presence of isolated leptons
- Prepare input** for the classifiers
 - Produce multiple datasets
 - Raw data (list of particles)
 - High Level features
- Store results** in parquet files
 - Dev. dataset (100k events)
 - Full dataset (5M events)



- Model #1
- Model #2
- ...

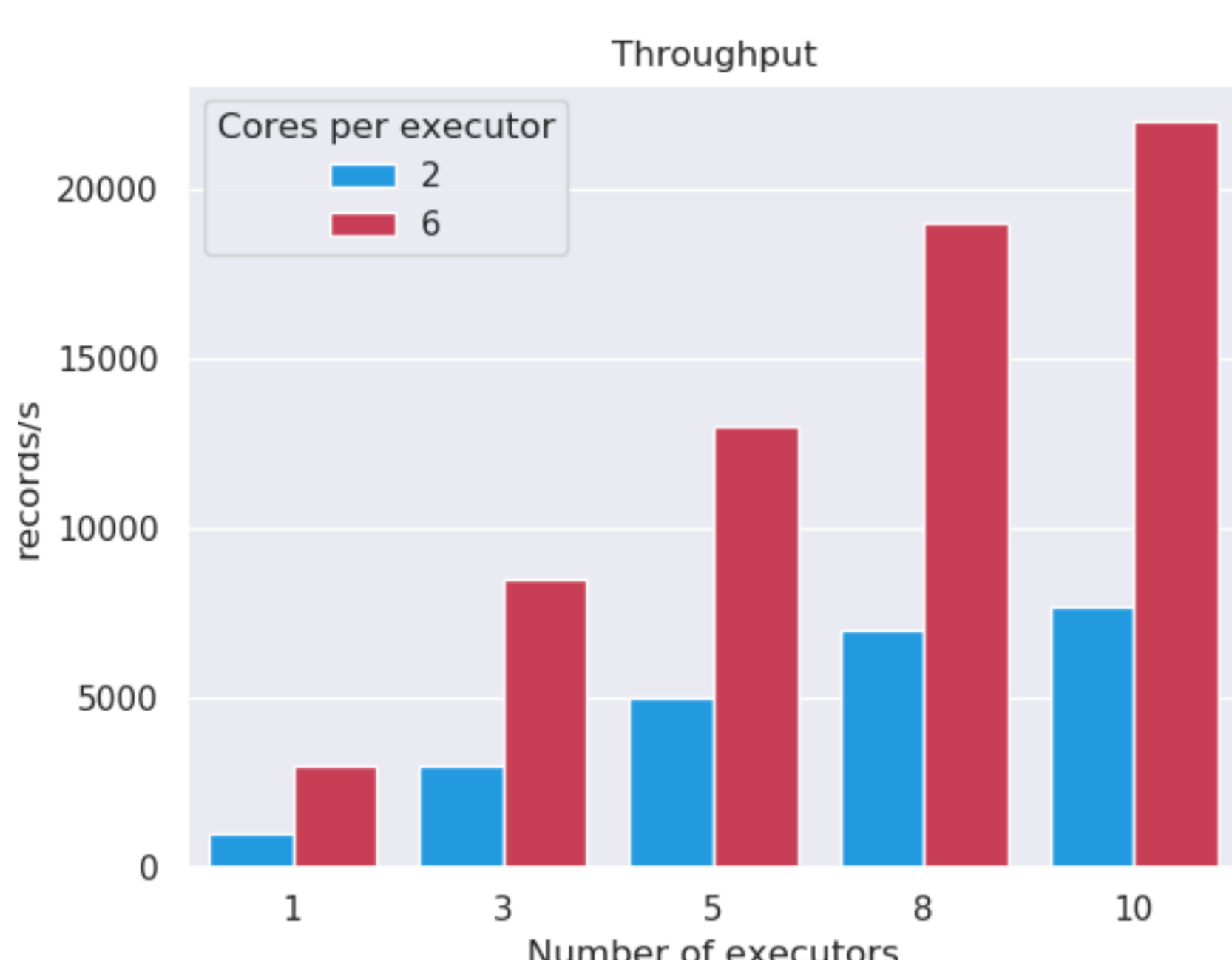
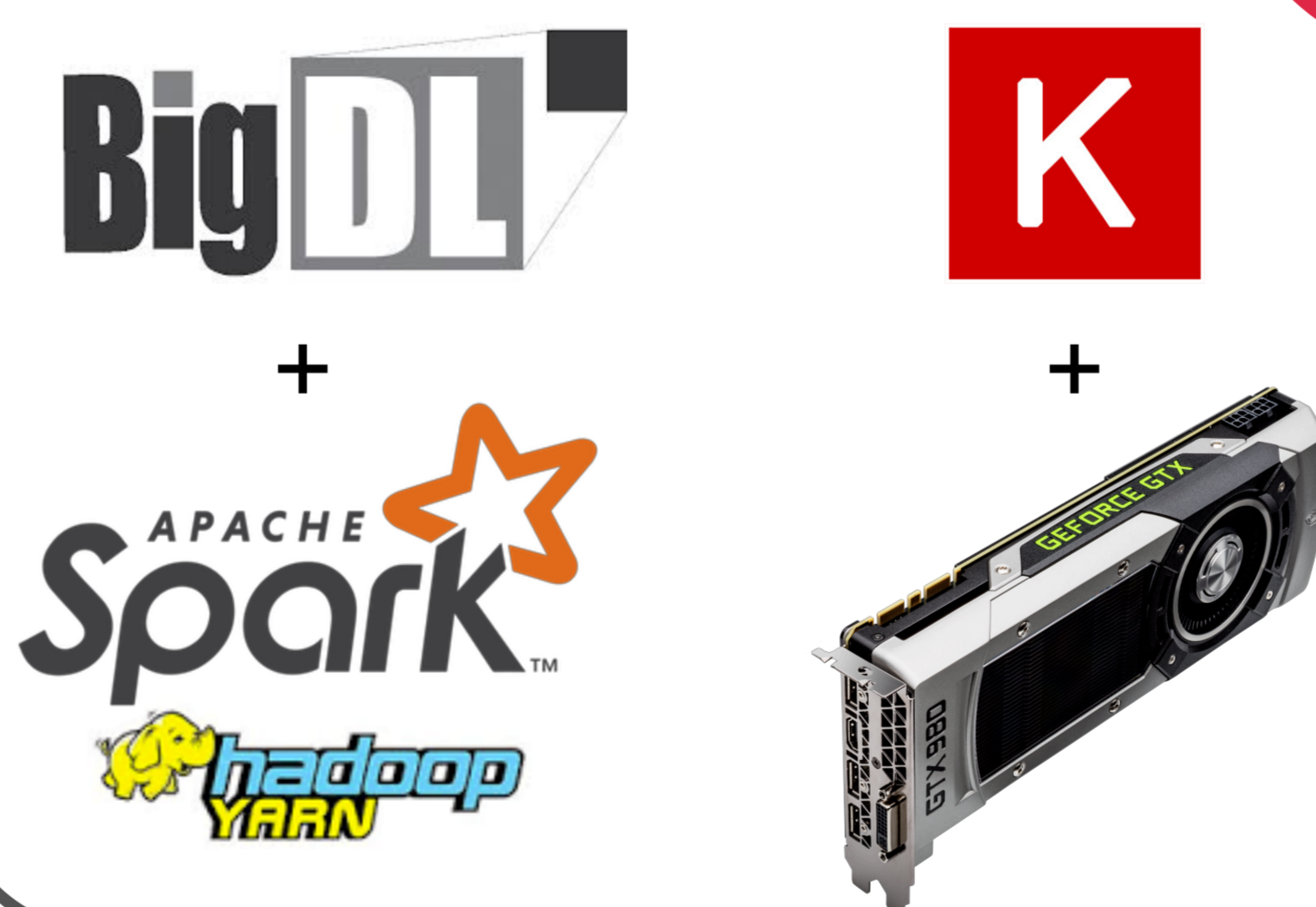
Parameters Tuning

Best Model

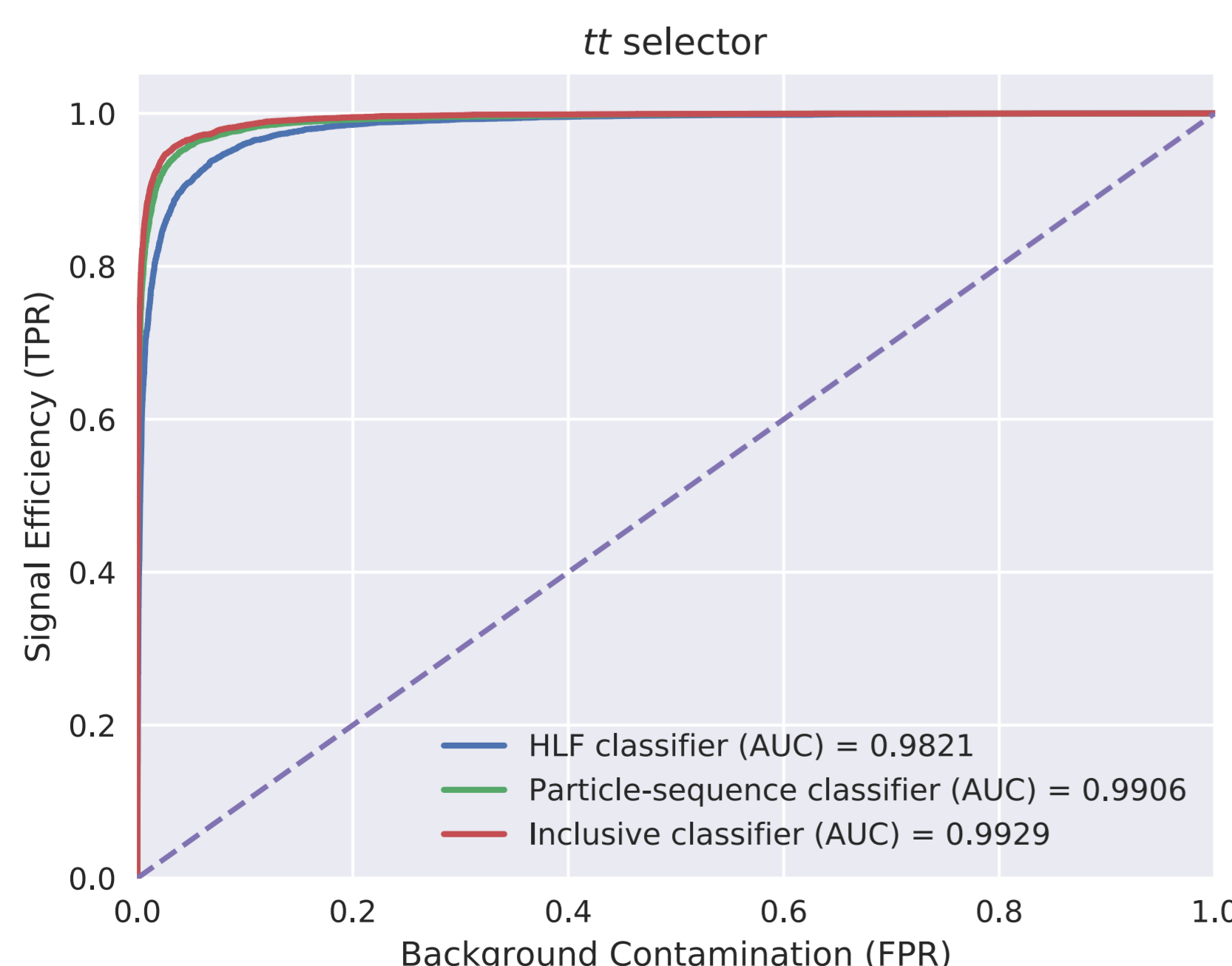
- Scan a grid of parameters to find the **best model**
- Train multiple models** at the same time (one per executor)

Training

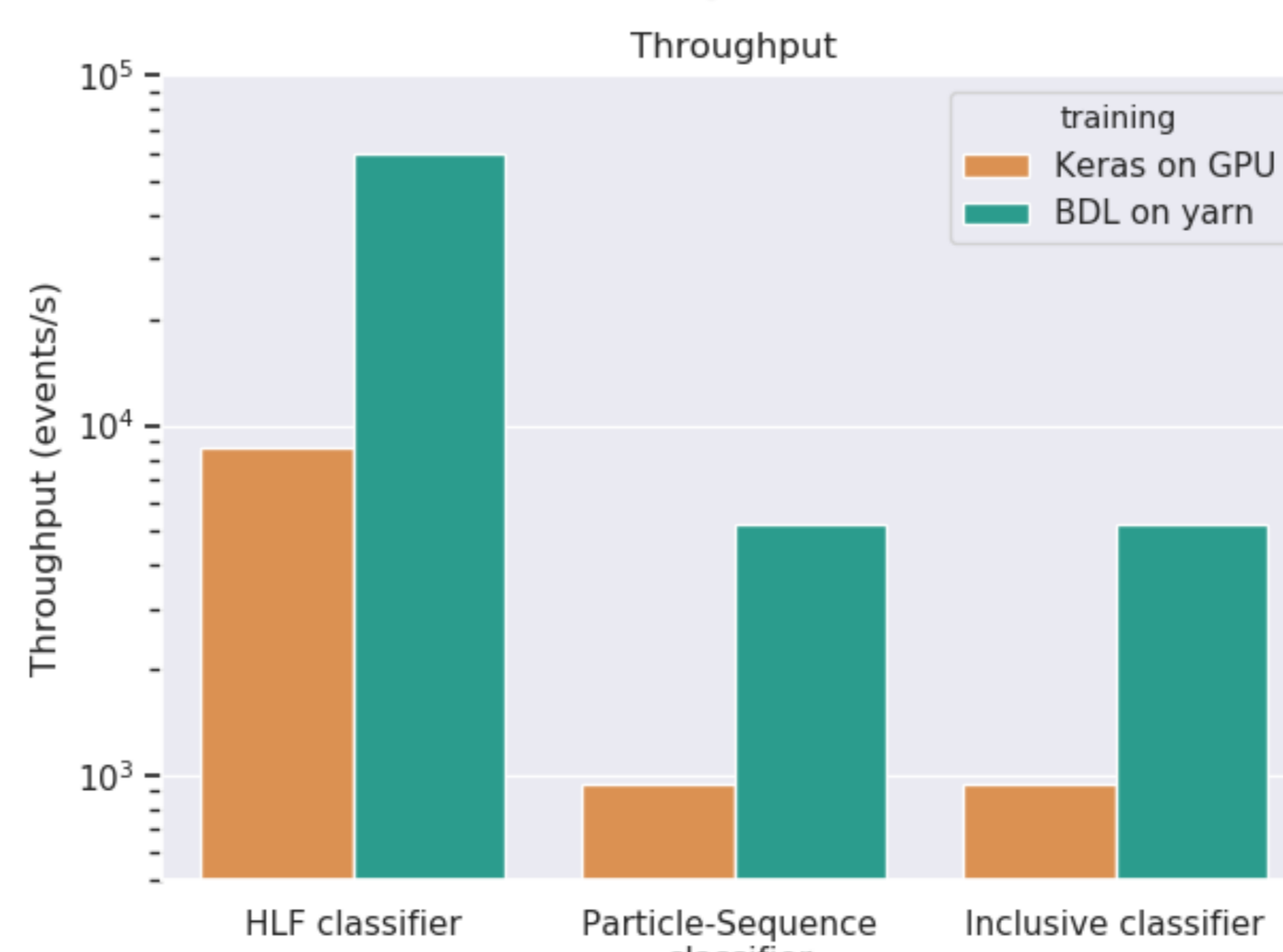
Trained the **three models** using various hardware and configurations



Observed near linear scalability of Intel **BigDL**



Reproduced the classifiers **performance** of the source paper



Throughput test measurements on the three different training methods and model types

Summary

- Created an end-to-end ML pipeline using Apache Spark
 - Python & Spark** allow to distribute computation in a simple way
 - Intel **BigDL** scales well and it is easy to use because it has a similar API to Keras
 - Interactive analysis made easier by **Jupyter Notebooks**
- Future work
 - Test pipeline using **cloud resources**
 - Further **performance improvements** on data preparation and training
 - Model Serving: implement inference on **streaming data**

