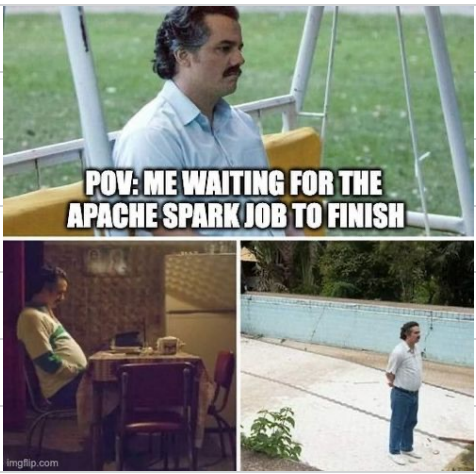
Session 10

Apache Spark

Jan 29, 2024



* Why can't we use Pandas for TBs of data?

(1) Memory problem (load everything in Ram)

(2) Single Core

100 mb csv File → 1GB - 2GB

1GB CS - File → 5GB of Ram space

A100 instance → $3.4

GPT-4 → OPen ai use

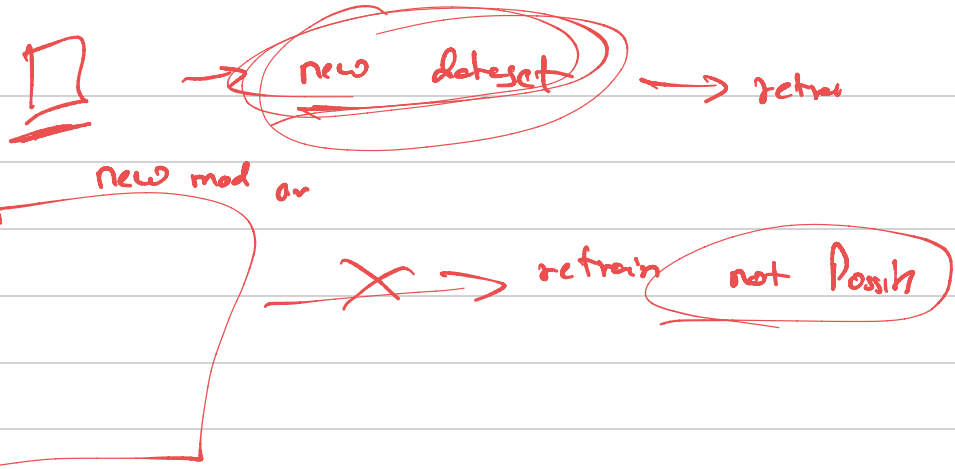
25,000 A100 instances

512 GB Ram
80 GB VRAM

1 - 2 months

$ million

$125 million



→ Why does it take more memory?

① Plain-text (CSV)

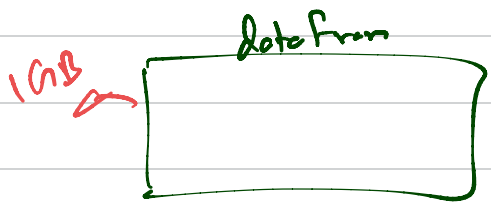
load it up

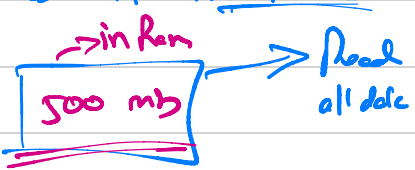
Pandas reads it → Complex struct

indexes, data type information

500 mb → CSV

Read it in Pandas





1GB + 500 mb → 1500 mb of space

→ What can we do to optimize this.

1. Read in chunks
2. Convert int64 → int32
3. Categorical data → objects

category dtype → consume lot less memory

Swap space → RAM → 4GB

RAM → swap space → HDD memory

Reading / writing → binary format

>> more efficient than CSV

→ DASK → why do we need it??

→ open source python source

(i) Chunk - base processing.

* Process data in chunks → Sequentially
→ Parallely

9 months → 1 baby

2 women → 4-5 months

(ii) Lazy Evaluation.

(iii) Distributed Computing → distribute task → all the cores of 4 cores

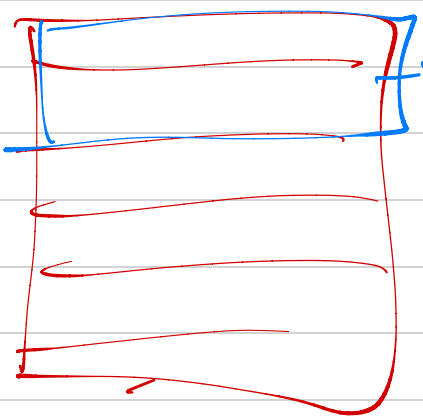
within a computer

Pandas → >> ① ② Setup functionality

③ Easier to Use

Not very common

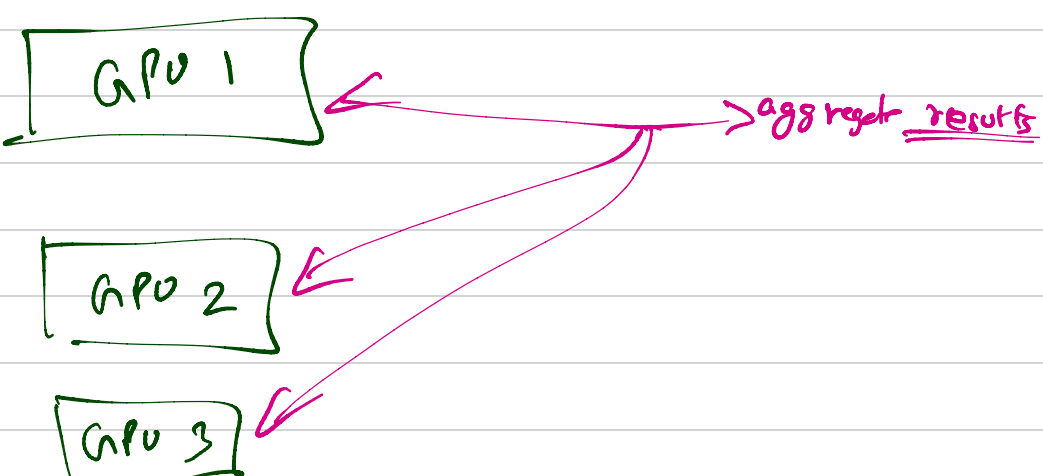
① Small data -> Pandas
② Spark



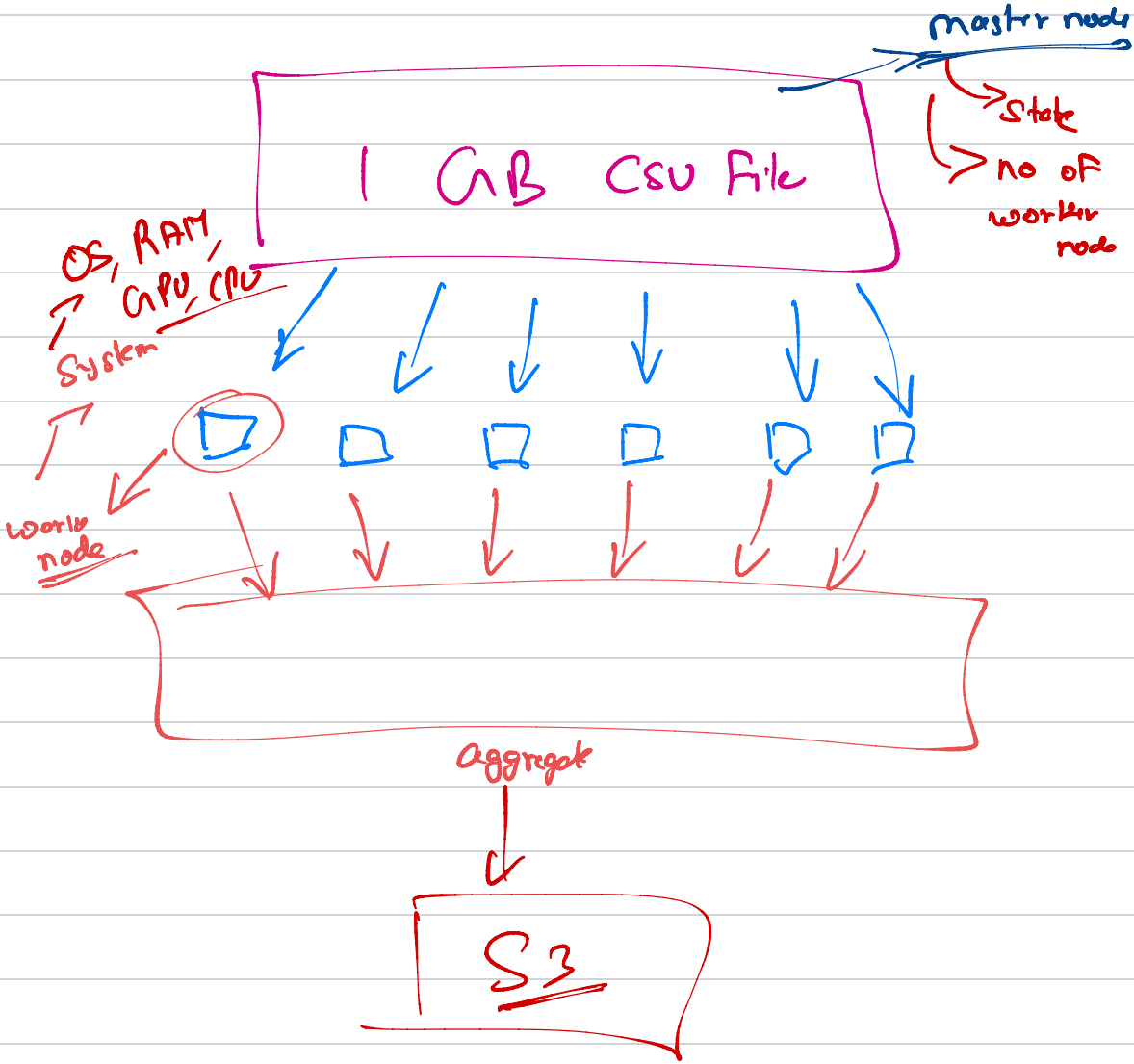
chunksize = 2 in Pandas

Dask = 2

TensorFlow



Spark = Why we need ?!



Why we use Spark

1. Distributed Computing
2. Open Source
3. lazy Evaluation
4. Not in memory.

ms. 16x lore → 100 worker nov
2.8x lers

range(1, 1000000)

For i in rang - - -)
i = 4 → in my ran

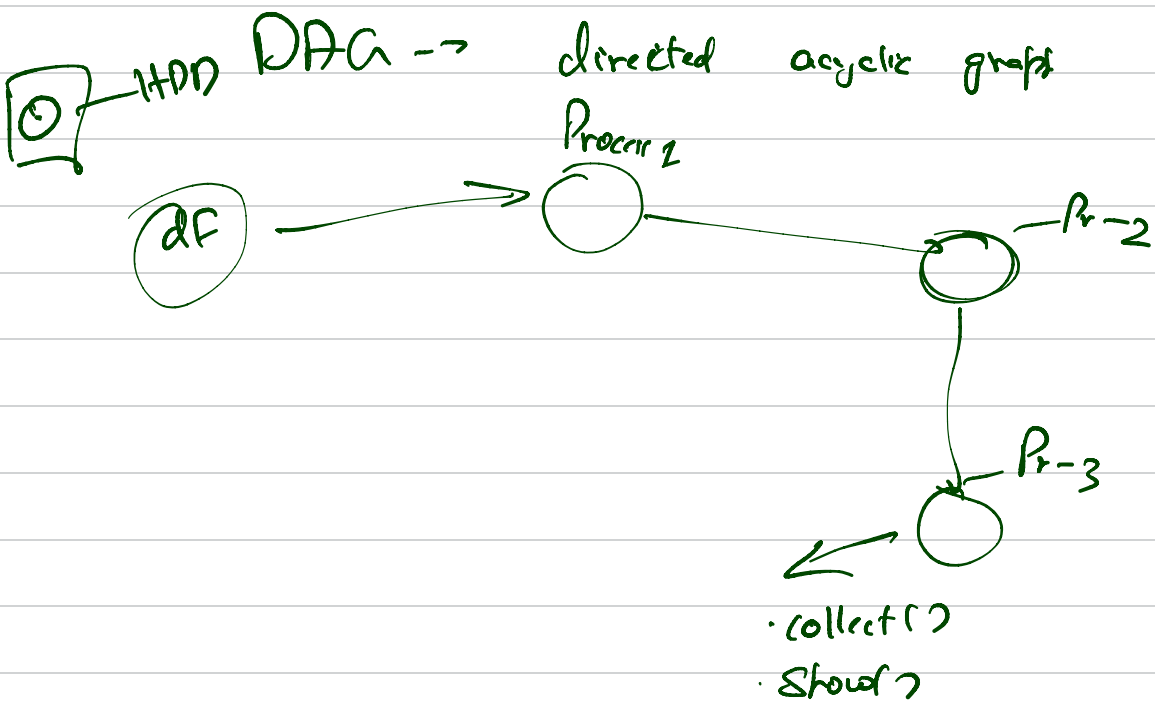
dF → width, height, length

width

yield - (generator function)

Why we need lazy evaluation:

(1) Efficient in computation



Spark -> SQL - spark

-> Mllib

-> xgboost

-> random forest