Estimate the resources required for your pyspark job?

2 main factors
===============

- Amount of data you are dealing with
- how quickly you want your job to complete

Note - Its not always the case that your job will complete faster if you give more resources

2 ways to allocate the resources

- static allocation
  the values are given as part of spark-submit

- dynamic allocation
  the values are picked on the requirement
  size of data, amount of computation needed
  resources are released after use.
  This helps the resources to be reused for other applications

  -initial executors
  -min executors
  -max executors
  -spark.dynamicAllocation.schedulerBacklogTimeout
  -spark.dynamicAllocation.executorIdleTimeout

Its not that you can estimate the resources at the very first time

you might start with a certain amount of resources based on your calculations, then you will run the job and monitor the job using spark UI and make changes based on the observations.

Scenario-1
===========

Lets say you have a 100 node cluster

each node is 16 cpu cores / 64 GB RAM

per executor the idle number of cpu cores - 5 cores

1 cpu core / 1 gb ram - background process

15 cpu cores / 63 gb ram

3 executors - 5 cpu cores / 19 GB RAM

100 nodes * 3 executors = 300 executors

(5 cpu cores / 19 gb ram)

300 mb reserved
40% - user memory
60% - unified region (execution + storage)
50% 50%

each cpu core - 1 gb for execution / 1 gb for storage

partition size is 128 mb how much execution memory is required to handle this partition?

data in memory takes more space than on disk

you can consider if you are processing a partition of size X

then you would require 4X-6X of memory on a normal side

128 mb partition on disk - 500 mb / 750 mb memory per cpu core

16 cpu cores / 64 gb ram

each node is 32 cpu cores / 64 gb ram

6 executors - 5 cpu cores / 10 gb ram

each cpu core = 2 gb ram (500 mb for execution / 500 mb for storage)

128 mb partition

64 mb / 256 mb (performance issues)

100 nodes (300 executors)

Executor - 5 cpu cores / 19 gb ram

you have a orders dataset - 100 gb

128 mb - 800 inital partitions

how much each customer has spent

aggregation based on customerid

orderid, orderdate, customerid, orderstatus, amount
1,   1st jan,   101,   closed,   100

2 stages
=========

stage 1 - 800 initial partitions (128 mb)
800 cores / 5

160 executors so that in this stage all of the tasks can execute in parallel.

stage 2 - the number of shuffle partitions is set to 200 by default

100 gb data / 200 partitions

on an average each partition will hold 500 mb data

4x to 6x (more than 2 GB of execution memory per cpu core)

600 cpu cores will sit idle looking that job is struggling with memory related issues...

make the number of shuffle partitions to 800 in this case

128 mb of data

ideal scenario in this case is 160 executors - 5 cpu cores / 19 gb ram

change the number of shuffle partitions to 800

in this case we will get everything done in parallel.

===================

80 executors (400 cpu cores)

1st stage - 800 initial paritions (128 mb each)
2 batches

2nd stage - 100 gb data

TRENDYTECH UPLIFT YOUR CAREER!

200 shuffle paritions - 500 mb per partition

400 shuffle partitions - 250 mb per partition

1 gb execution / 1 gb storage

here the 1st stage is done in 2 batches
2nd stage is done all in parallel

80 executors / change the number of shuffle partitions to 400

===============

40 executors (200 cpu cores)

1st stage - 800 partitions
4 batches

2nd stage -

100 gb / 200 = 500 mb / partition

300 shuffle partition / 400 shuffle partition

350 mb   / 250 mb

even if we are not caching anything which of this is recommended

300 shuffle partitions

200 cores

2 batches

=============

recommend this

400 shuffle partitions (250 mb)

200 cores

2 batches

===============

TRENDYTECH UPLIFT YOUR CAREER!

201 shuffle partitions

2 batches

scenario 2
============

Joins (customer id)

orders / customers
100 gb 50 gb

orders - 100 gb (800 partitions - 128 mb)
customers - 50 gb (400 partitions - 128 mb)

1200 cpu cores to get the 1st stage done in parallel

240 executors (5 cores)

stage 1 - 240 executors (1200 cores)

stage 2 - 150 gb data (200 shuffle partitions)

each partition will hold (750 mb data)

here we could have 1200 shuffle partitions (128 mb data)

=============

120 executors (600 cpu cores)

1st stage -

orders - 100 gb (800 partitions - 128 mb)
customers - 50 gb (400 partitions - 128 mb)

2 batches

2nd stage - 150 gb

600 shuffle partitions - 250 mb (1 gb / 1gb for storage)

============

60 executors (300 cpu cores)

1st stage -

orders - 100 gb (800 partitions - 128 mb)
customers - 50 gb (400 partitions - 128 mb)

4 batches

2nd stage - 300 shuffle partitions

150 gb / 300 = 500 mb

600 shuffle partitions

2nd stage will be done in 2 batches (250 mb)

===========

30 executors (150 cpu cores)

1st stage -

orders - 100 gb (800 partitions - 128 mb)
customers - 50 gb (400 partitions - 128 mb)

8 batches

2nd stage - 200 shuffle partitions

750 mb per partition

300 shuffle partitions - 2 batches (500 mb/partition)

450 shuffle partitions - 3 batches (350 mb/partition)

600 shuffle partitions - 4 batches (250 mb/partition)

=============

Driver

ideally driver is not responsible for any of the processing

TRENDYTECH UPLIFT YOUR CAREER!

it just manages things like creating execution plan, collect, broadcast

5 cpu / 19 gb ram

2 cpu cores / 8 gb ram

========

how much data?

is it a long running job?

complexity of the job

whether you are caching or not

how quickly you want your job to run

How to get Interview calls?

Resume
Naukri
LinkedIn
Referrals

Resume
========

Skills - Pyspark, azure databricks, azure datafactory, HDFS, azure datalake storage, azure synapse, git & github, pytest, agile, performance tuning

Big Data
pyspark
Data Engineering
Azure Cloud
SQL
Hadoop
Azure Databricks
Apache Spark
Azure Datafactory
Agile
Azure Datalake Storage
DWH
python

TRENDYTECH UPLIFT YOUR CAREER!

ETL

Projects/work
================

=> build a data pipeline that ingest data to Azure Datalake/HDFS from multiple data sources

=> led the migration of data from multiple sources to ADLS gen2 thereby empowering teams to analyze it.

=> Led the migration to cloud that resulted in a cost saving of 37%

=> Led the performance initiatives that optimized the queries to yield a 70% improvement in data processing speed.

=> Rather than analysing data in a database, move this data to a datalake and analyse it.

=> I was involved in migrating one project from Hive to Pyspark

=> Migrating a project from pyspark to databricks

=> wrote a bunch of transformations in one of the existing pyspark project

=> used serverless synapse for one of the daily jobs and optimized it to scan less data by using columnar file formats and other techniques.

=============

Summary

Skills

Work Experience

Education

Projects (fresher)

Achievements

cerficates

TRENDYTECH UPLIFT YOUR CAREER!

Languages

https://www.beamjobs.com/resumes/data-engineer-resume-examples

======================

your resume should never be more than 2 pages..

recommended is 1 page (2 columns)

2 pages (single column)

================

If you are a fresher

if you are someone with under 3 years of experience

Database, SQL.. Big Data, Pyspark, Datbricks...

over 3 years of experience upto 8 years

last 3.5 years you are in big data

if you have more than 8 years experience..

for last 4 years you are working on Big Data

ETL, Database, SQL, Java, Python

Mainframes, testing

last 3/3.5 years show big data related experience

ETL, Database, SQL, Java, Python

===============================

How to Optimize your Resume
===============================

ATS (application tracking system)

Naukri Profile

TRENDYTECH
UPLIFT YOUR CAREER!

===============

completeness of your profile

100% complete

upload ATS compliant resume

1 month notice period

Serving notice period | Immediate joiner

update your naukri profile regularly

apply for jobs regularly

reply to any inbound mails, messages

log in to the platform daily

write a blog or a linkedIn article and provide the link.

How to Get Calls from LinkedIn
================================

create a linkedIn profile & it should be complete in all aspects

100% complete

you want more job calls!

when creating profile...

1. search (100% complete, you put relevant keywords)

big data developer
data engineer
senior data engineer
big data architect
data engineering architect
data engineering manager

SEO (search engine optimization)

TRENDYTECH UPLIFT YOUR CAREER!

2. make it visually appealing so that people visit your profile

3. when someone is on your profile, there should be a way for them to connect with you..

email in your contact
resume is attached..

============================

for keeping your profile active

I suggest you post 1 technical post in a week.

write it in a neat and concise way..

#bigdata #dataengineering

============

you can send connection requests to people in data engineering

500+

==============

if someone post good content around big data

you can like it, you can post some meaningful comment on it.

you can repost

repost, like, comment

================

1. recruiters approach you

2. you apply from the jobs portal

Easy Apply

Referrals

TRENDYTECH UPLIFT YOUR CAREER!