**به نام خدا**

**علی عباس حقیقت**

**220797069**

**موضوع بخش اول تمرین اول درس داده کاوی :**

**Spark VS Hadoop**

با گسترش روز افزون اینترنت و در نتیجه آن گسترش حجم دیتای تولید شده در محیط هایی هم چون شبکه های اجتماعی – اینترنت اشیا – اینترنت کلاسیک و در کل اطلاعات تولید شده در هر بخشی از جمله تحقیقات علمی و آماری و ... (به عنوان مثالی برای سرعت و حجم تولید داده به گزارش مجله معتبر Forbes ، 90 درصد داده جهان در دو سال اخیر تولید شده است)

باعث شده تا علم مطالعه داده ها با توجه به حجم بسیار زیاد و سرعت تولید فراوان و همچنین اهمیت بحث امنیت داده موجب شده تا سیستم ها و روش های داده کاوی توزیع شده که از جمله ی آنها hadoop و spark هستند ، توسعه و محبوبیت بسیاری بیابند .

همانطور که میدانیم اسپارک و هدوپ که هر دو از محصولات شرکت apache هستند وظیفه مدیریت پردازش اطلاعات را در داخل 1 تا n تا سیستم مختلف را به صورت توزیع شده به عهده دارند همچنین میدانیم هردو امکان پشتیبانی از انواع داده چه به صورت ساختاری چه بدون ساختار را دارند همچنین میدانیم هردو به طرزی از ایده mapreduce استفاده میکنند (هدوپ مستقیم تر و اسپارک به صورت فرعی) ، همچنین در زمینه استفاده از الگوریتم های هوشمند و به طور ویژه یادگیری ماشینی و پشتیبانی از انواع پایگاه داده و مدلهای نموداری – آماری نیز باهم شباهت دارند .

اما چه امری باعث ایجاد تفاوت های فاحش بین سرعت عملکرد این دو فریم ورک محبوب (اسپارک و هدوپ ) شده است تا حدی که در بعضی منابع به سرعت 100 برابری اسپارک نسبت به هدوپ اشاره میشود .

مقایسه ساختار :

هدوپ در ساختار خود از سیستم فایل توزیع شده HDFS بهره میبرد ، همچنین از YARN برای مدیریت خوشه ها(کامپیوتر ها) استفاده میکند بحث پردازش موازی را نیز با استفاده از تکنولوژی mapreduce (متعلق به شرکت گوگل) مدیریت میکند . همچنین برای یادگیری ماشینی از موتور submarine بهره میبرد .

همچنین اسپارک نیز در ساختار خود ماژول های زیادی در زمینه کار با دیتا بیس ها و بحث مرتب سازی داده ها و همچنین یادگیری ماشینی دارد (MLLib) که علاوه بر اینکه از بسیاری از قابلیت های زبان های میزبان(جاوا- اسکالا-پایتون-R) خود به ویژه پایتون در زمینه یادگیری ماشینی و data cleaning و data manipulation استفاده میکند بلکه در بعضی موارد مانند بحث یادگیری ماشینی از ماژول ویژه ای مثل (MLlib) بهره میبرد که شباهت با sklearn دارد . همچنین Graphx هم قابلیت انجام محاسبات بر بستر gpu را ممکن میسازد

همچنین به لحاظ تحمل خطا و منبع باز بودن هم هردو فریم ورک شرایطی شبیه به یکدیگر دارند ، یعنی هردو منبع باز و با قابلیت مدیریت خطا های احتمالی هستند .

اما تفاوت های اصلی بین دو سیستم در جایی شروع میشود که میخواهیم سرعت آن ها را با یکدیگر مقایسه کنیم در اینجا متوجه میشویم اسپارک با توجه به ساختار الگوریتمی خاصی که دارد و به نحوی طراحی شده است که اطلاعات مورد پردازش را به صورت بسیار بهینه در RAM سیستم ذخیره کند فلذا بسیار سریع تر از هدوپ است ولی هدوپ هم به لخاظ اینکه برای استفاده از دیسک مناسب است میتواند برای دیتا های حجیم تر در بعضی موارد مورد ترجیح باشد اما همین سرعت کم باعث میشود برای پردازش realtime اطلاعات گزینه مناسبی برای ما نباشد و درست در مقابل اسپارک دراین زمینه قرار بگیرد .

در حقیقت میتوان گفت مهمترین نقطه ضعف هدوپ که همان processing linear حجم بسیار عظیم اطلاعات است در که در مقابل iterative realtime processing بسیار سریع اسپارک قرار میگیرد ، در حقیقت مهمترین برتری آن نیز هست چرا که قابلیت پردازش دیتای حجیم را به آن میدهد

همچنین باید اضافه کنیم هر دو فریم ورک از ETL برای یک پارچه سازی داده های خود بهره میبرند.

همچنین در مقایسه مقیاس پذیری هدوپ بسیار ساده است فقط کافی است به تعداد دلخواه به کامپیوتر هایمان اضافه کنیم اما اسپارک به دلیل رم بیس بودن مقداری سخت تر است .

به لحاظ ایمنی هدوپ فوق العاده امن است چرا که از  LDAP, ACLs, Kerberos, SLAs, پشتیبانی میکند . ولی اسپارک از ایمنی بالایی برخوردار نیست بلکه به بعضی ماژول های مشترکش با هدوپ تکیه می کند.

به لحاظ یادگیری ماشینی هم با توجه به برتری ویژه اسپارک در سرعت از هدوپ در این بخش بهتر عمل میکند

در یک کلام برای جمع بندی میتوان گفت برای اپ های کوچک با حجم داده کمتر و همچنین نیاز به پردازش realtime اسپارک مناسب است. اما برای محاسباتی که به لحاظ زمانی در تنگنا نیستن و حجم داده هم فوق العاده زیاد است و یا از انواع داده های تصویری با حجم زیاد استفاده شده است . هدوپ کارایی بهتری دارد.

همچنین پیاده سازی زبان اسپارک ساده تر است و زبان های بیشتری را نسبت به هدوپ (فقط پایتون و جاوا برای mapreduce ) پشتیبانی میکند .

|  |  |  |
| --- | --- | --- |
| فاکتور مورد بررسی | اسپارک | هدوپ |
| سرعت در پردازش | تا حدود 100 برابر سریعتر  از هدوپ | سرعت پایین اما ویژگی های مهم دیگری دارد |
| پردازش در انواع حافظه | سعی در پیاده سازی حداکثری در رم را دارد . io حداقل | بیشتر استفاده از دیسک و io بالا برای هندل دیتای بزرگ |
| پردازش به شکل توزیعی | ذاتا برای پردازش توزیعی توسعه داده شده | ذاتا برای پردازش توزیعی توسعه داده شده |
| پردازش دیتای حجیم | برای پردازش حجیم مناسب نیست چون ویژگی اصلی آن استفاده حداکٍری از رم و سرعت بالا است | ویژگی و برتری اصلی آن پردازش در حجم بالا وسرعت کم با اتکا به دیسک است |
| پردازش realtime | به علت سرعت بالا پردازش در لحظه را هندل میکند | برای پردازش در لحظه بیش از حد کند است |
| ایمنی | ایمنی مناسبی ندارد | در بالاترین سطح امنیت است |
| تحمل خطا و مقیاس پذیری | تحمل خطا بالا ولی به لحاظ مقیاس پذیری به دلیل اتکا بیش از حد به رم کم | تحمل خطا بالا همچنین مقیاس پذیری بسیار بالا و ساده |
| منبع باز بودن | منبع باز است | منبع باز است |
| سادگی در پیاده سازی زبان | پیاده سازی ساده تر و همچنین از زبان های بیشتری پشتیبانی میکند | پیاده سازی اندکی سخت تر فقط از دو زبان پشتیبانی میکند |

**منابع : منابع زیر صرفا مطالعه شده اند نه کپی برداری**

[Hadoop vs Spark: Detailed Comparison of Big Data Frameworks (phoenixnap.com)](https://phoenixnap.com/kb/hadoop-vs-spark)

[دیجی دانش | مقایسه Hadoop و Spark (digidanesh.org)](https://digidanesh.org/blog/78/%D9%85%D9%82%D8%A7%DB%8C%D8%B3%D9%87-Hadoop-%D9%88-Spark)

Forbes

[How Does Spark Use MapReduce? - DZone Big Data](https://dzone.com/articles/how-does-spark-use-mapreduce#:~:text=Apache%20Spark%20does%20use%20MapReduce%20%E2%80%94%20but%20only,the%20idea%20of%20it%2C%20not%20the%20exact%20implementation.)

[Machine Learning With Spark. A distributed Machine Learning… | by MA Raza, Ph.D. | Towards Data Science](https://towardsdatascience.com/machine-learning-with-spark-f1dbc1363986)

[تفاوت های عمده Hadoop و Spark در چیست | ۲۰۱۹ (classicfoxvalley.com)](https://fa.classicfoxvalley.com/collate/what-are-the-major-differences-hadoop-and-spark-b9dffd/)

[Spark vs. Hadoop MapReduce: Which big data framework to choose (scnsoft.com)](https://www.scnsoft.com/blog/spark-vs-hadoop-mapreduce#:~:text=Linear%20processing%20of%20huge%20datasets%20is%20the%20advantage,In%20many%20cases%20Spark%20may%20outperform%20Hadoop%20MapReduce.)