

## SENSOR MA'LUMOTLARINI QAYTA ISHLASHDA TAQSIMLANGAN TEXNOLOGIYALAR TAHLILI

**Barno Rustamovna Annazarova**

Muhammad al-Xorazmiy nomidagi TATU Urganch filiali magistranti

[barno220517@gmail.com](mailto:barno220517@gmail.com)

### ANNOTATSIYA

Ushbu maqolada sensor ma'lumotlari manbalari, ularni qanday shakllanishi va qayta ishlashga qanday tayyorlanishi haqida so'z boradi. Ma'lumotlarni qayta ishlashda bir necha texnologiyalar mavjud. Lekin sensor ma'lumotlari turli xil bo'lishi va real vaqtda to'xtovsiz bo'lishi ularni katta hajmli ma'lumotlar deb atashga sabab bo'ladi. Shuning uchun ham **big data** larni qayta ishlashda taqsimlangan hamda parallel texnologiyalardan foydalanish kerak bo'ladi. Bu maqolada bir necha taqsimlangan texnologiyalar tahlil qilingan.

**Kalit so'zlar:** sensor ma'lumotlari, big data, Hadoop, MapReduce, Apache Spark, qayta ishlashga tayyorlash.

### ABSTRACT

This article discusses the sources of sensor data, how they are formed, and how they are prepared for processing. There are several technologies involved in data processing. But the fact that sensor data is diverse and non-stop in real time is the reason to call it big data. Therefore, it is necessary to use distributed and parallel technologies in the processing of big data. Several distributed technologies are analyzed in this article.

**Keywords:** sensor data, big data, Hadoop, MapReduce, Apache Spark, preparation for processing.

### KIRISH

«Big Data» atamasi 2011 yildan beri mashhur bo'la boshladi. Bugun har bir kishi kamida bu termini bir marta eshitgan. Texnologiyaning rivojlanishi bilan ma'lumotlar miqdori geometrik progressiya tarzida o'sishni boshladi.[3] An'anaviy vositalar axborotni qayta ishlash va saqlashga bo'lgan ehtiyojni qoplashni to'xtatdi. Yuzlab terabaytdan oshadigan va doimiy ravishda o'sib borayotgan ma'lumotlarni qayta ishlash uchun maxsus algoritmlar yaratilgan. Ular «big data» deb nomlanadi. Bugungi kunda ma'lumotlar turli manbalardan juda katta hajmda to'planadi: Internet, aloqa

markazlari, mobil qurilmalar va boshqalar. Ko'pincha bunday ma'lumotlar aniq tuzilishga va tartibga ega emas, shuning uchun odam uni biron bir faoliyat uchun ishlata olmaydi. Tahlilni avtomatlashtirish uchun katta ma'lumotlar texnologiyalari qo'llaniladi.

## ADABIYOTLAR TAHLILI VA METODOLOGIYA

Katta hajmli ma'lumotlarni qayta ishlashga tayyorlash va uni qayta ishlash bir muncha murakkab vaziyat hisoblanadi. Sensor ma'lumotlari doimiy ravishda uzluksiz qabul qilinadi va qayta ishlanadi. Shuning uchun ham ularni ananaviy texnologiyalar yordamida qayta ishlashning imkoniyati yo'q. Bu maqolada sensor ma'lumotlarini qayta ishlashda parallel texnologiyalardan foydalanish hamda ularning tahlili haqida fikr bildirilgan. Ma'lumotlarni qayta ishlashda bir necha taqsimlangan texnologiyalar mavjud. Bu yerda ko'proq MapReduce va Apache Spark texnologiyalari tahlil qilingan.

**Apache Spark** - bu katta ma'lumotlarda klasterli hisoblash va katta hajmdagi ma'lumotlarni qayta ishlash uchun ishlatiladigan platforma. Spark tezkor xotiradagi ma'lumotlarni qayta ishlaydi va doimiy xotiraga kamdan-kam murojaat qiladi, shuning uchun u boshqa texnologiyalarga nisbatan juda tez ishlaydi.

### Apache Spark Core

Bu butun platformaning asosini tashkil etuvchi asosiy ma'lumotlarni qayta ishlash mexanizmi. Yadro saqlash tizimlari bilan o'zaro ta'sir qiladi, xotirani boshqaradi, klasterdagi yukni rejalashtiradi va taqsimlaydi. Shuningdek, u dasturlash tillarining API-ni qo'llab-quvvatlash uchun javobgardir.

### Spark SQL

Ushbu modul tuzilgan ma'lumotlar bilan ishlashni soddalashtirish uchun ishlatiladi va SQL tilida so'rovlarni bajarishga imkon beradi. Uning asosiy vazifasi ma'lumotlar muhandislari ma'lumotlarni saqlashning taqsimlangan tabiati haqida o'ylamasliklarini, balki ulardan foydalanish stsenariylariga e'tibor qaratishlarini ta'minlashdir.

### Streaming

Real vaqtda oqimli ma'lumotlarni qayta ishlashni kengaytiriladigan, yuqori unumdorlik va xatolarga chidamliligini ta'minlaydi. Kafka, Flume, Kinesis va boshqa tizimlar Spark uchun ma'lumot manbalari sifatida harakat qilishi mumkin.

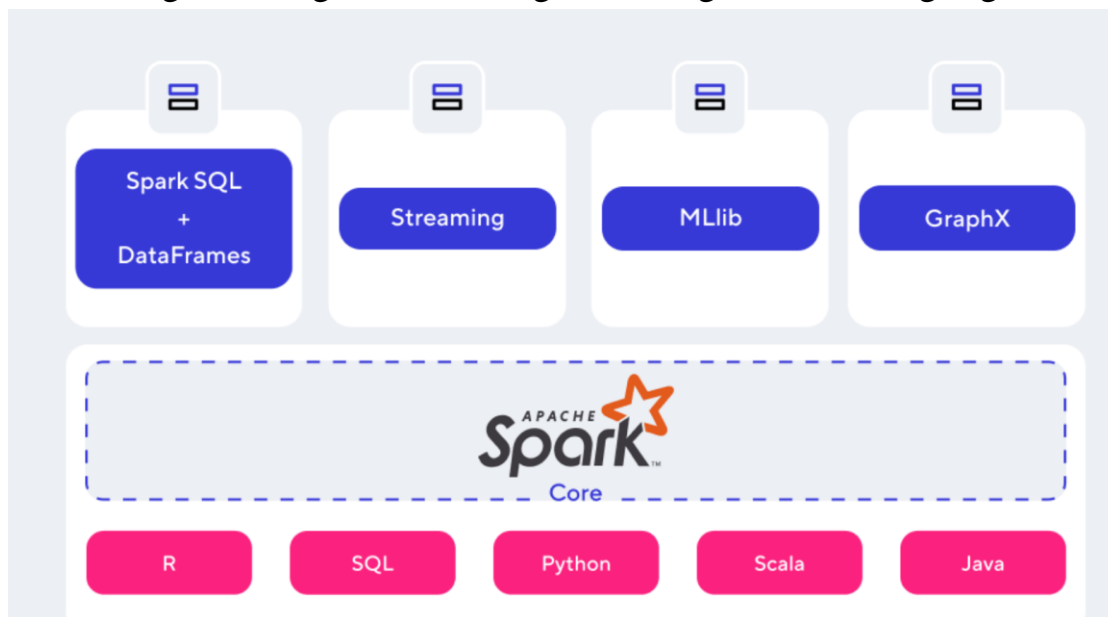
### MLib

Bu kengaytiriladigan past darajadagi mashinani o'qitish kutubxonasi. Kutubxonada klasterlash, regressiya, tasniflash va

hamkorlikda filtrlash kabi turli xil mashinalarni o'qitish algoritmlari mavjud.

## GraphX

Grafiklar ustida manipulyatsiyalar va ularga parallel ishlov berish uchun xizmat qiladi. GraphX grafik ulanishini, daraja taqsimotini, o'rtacha yo'l uzunligini va boshqalarni o'lchashi mumkin. Shuningdek, u grafiklarni ulashi va ularni tezda o'zgartirishi mumkin. Grafiklar bo'yicha o'rnatilgan operatsiyalardan tashqari, GraphX ham PageRank algoritmini amalga oshiradigan kutubxonaga ega.



1-rasm. Spark kutubxonalari

## NATIJALAR

### Apache MapReduce va Apache Sparkni taqqoslash.

Ilgari MapReduce odatda katta ma'lumotlarni qayta ishlash uchun ishlatilgan. Bu Big Datada ma'lumotlarni qayta ishlash uchun asos yaratgan Hadoop komponenti. Ammo uning ikkita global muammosi bor.

**Kam unumdorlik.** MapReduce ikki bosqichda hisob-kitoblarni amalga oshiradi. Birinchidan, u ma'lumotlarni qismlarga ajratadi va ularni qayta ishlash uchun klaster tugunlariga o'tkazadi. Keyin har bir tugun ma'lumotlarni qayta ishlaydi va natijani asosiy tugunga yuboradi, bu yerda yakuniy hisoblash natijasi hosil bo'ladi. MapReduce doimiy ravishda diskka kirishadi, u yerda hisob-kitoblarning barcha oraliq va yakuniy natijalarini saqlaydi. Shu sababli, u kechikishlar bilan ishlaydi, bu esa MapReducedan oqim ma'lumotlarini va mashinani o'qitish vazifalarini qayta ishlash uchun foydalanishni imkonsiz qiladi.

**Yuqori qiynchilik.** Yaxshi MapReduce yechimini yozish yuqori darajadagi tajribani talab qiladi. Hatto tajribali mutaxassis

ham osongina xatoga yo'l qo'yishi yoki samarasiz algoritm yozishi mumkin.

Ammo 2014 yilda yana bir vosita - Spark mashhur bo'la boshladi va endi u MapReduceni amalda siqib chiqardi. Apache Spark MapReducening afzalliklarini saqlab qolgan holda kamchiliklarini bartaraf etish uchun ishlab chiqilgan. MapReduce muammolarini qanday hal qilish mumkin:

- Spark xotiradagi ma'lumotlarni qayta ishlaydi va diskka deyarli kirishga imkon bermaydi. Agar qayta ishlanayotgan ma'lumotlar miqdori operativ xotira miqdoridan katta bo'lsa, Spark ma'lumotlarning bir qismini diskka ko'chiradi. Shu bilan birga, unga ko'plab optimallashtiruvchilar o'rnatilgan bo'lib, ular diskka kirish sonini kamaytirishga imkon beradi. Shuning uchun, Spark MapReduce'dan o'nlab, ba'zan yuzlab marta tezroq.

- Spark-da turli xil dasturlash tillari uchun API mavjud, shuning uchun kod yozish ancha oson va kodning o'zi ixchamroq. Odatda, ishlab chiquvchilar Spark-da juda yuqori darajadagi ko'rsatmalar yozadilar va ularni qanday qilib optimal tarzda bajarish kerakligini uning o'zi hal qiladi. Ko'pincha u buni erkakdan yaxshiroq qila oladi.

## XULOSA

○ Apache Spark - bu Big Data uchun ma'lumotlarni qayta ishlash tizimi. U RAMda ishlaydi va diskka kamdan-kam kirishadi, shuning uchun u ma'lumotlarni juda tez qayta ishlaydi.

○ Ilgari ma'lumotlarni qayta ishlashning de-fakto standarti Hadoop MapReduce edi. Ammo uning ikkita asosiy muammosi bor: past unumdorlik va yuqori rivojlanish murakkabligi.

○ Endi standart amalda Sparkga aylandi. U MapReduce ning afzalliklarini saqlab qolgan holda kamchiliklarini bartaraf etish uchun yaratilgan.

○ Apache Spark va katta ma'lumotlar bilan ishlash uchun boshqa texnologiyalar bulutlarda faol qo'llaniladi, chunki ular bulutli xizmatlarning barcha afzalliklarini olish imkonini beradi.

## REFERENCES

1. D. Ávila A, Papavasiliou N, Löhdorf Computational Parallel and distributed computing for stochastic dual dynamic programming. Management Science (2022) 19:199–226
2. T.K. Urazmatov, X.SH. Kuzibayev, B.. BNurmetova “ANALYSIS OF BIG DATA PROCESSING TECHNOLOGIES” MIP: Engineering-2020 - Modernization, Innovations, Progress:



Advanced Technologies in Material Science, Mechanical and Automation Engineering"

3. Otabek Khujaev, Omonboy Khalmuradov, Ortiq Ruzibayev, Shukurulloh Ismoilov, Khudayshukur Kuzibaev "Finding optimal architecture of neural networks for predicting referrals on the virtual museums website" 2021 International Conference on Information Science and Communications Technologies (ICISCT)

4. Nataliya Shakhovska, Nataliya Boyko, Yevgen Zasoba, Eleonora Benova, "Big Data Processing Technologies in Distributed Information systems" The 6th International Symposium on Emerging Information, Communication and Networks 7, 2019, (EICN 2019)

5. Kay R., Little S. Assessing the Fit of the Logistic Model: A Case Study of Children with Haemolytic Uraemic Syndrome // Applied Statistics. - 1986. - Vol. 35.-P. 16-30.

6. Hosmer DW, Lemeshow S. Applied logistic regression // Wiley-Interscience. 2000. - P. 528.

7. Langley P., Iba W., Thompson K. An analysis of Bayesian classifiers // In The Tenth National Conference on Artificial Intelligence. AAAI Press, San Jose. - 1992. - P. 399-406.

8. Burges CJC A tutorial on support vector machines for pattern recognition // Data mining and knowledge discovery. - 1998.No.2(2).-P

9. Zhang GP Neural Networks for Classification: A Survey // IEEE Transactions on Systems, Man, and Cybernetics-Part C: Applications and Reviews. - 2000. - Vol. 30, No 4. - P. 44-49.

10. Landwehr N., Hall M., Frank E., Logistic model trees // Machine Learning. - 2005. - No. 59(1). - P. 161-205.