NILTON KAZUYUKI UEDA

DATA ANALYTICS

FRAMEWORK DE BIG DATA

AULA 02

SUMÁRIO

O QUE VEM POR AÍ?

Durante essa aula, apresentaremos os conceitos fundamentais do Apache Spark e aprenderemos como realizar operações básicas de transformação e manipulação de dados em escala. Desde a leitura e gravação de dados até a aplicação de filtros, mapeamentos e agregações, você descobrirá como essas operações podem ser executadas de maneira eficiente e distribuída no ambiente Spark.

Ao longo do curso, você terá a oportunidade de praticar as operações básicas utilizando as linguagens de programação Scala ou Python, amplamente utilizadas com o Spark. Você aprenderá ainda a criar RDDs (Resilient Distributed Datasets), a estrutura fundamental do Spark, e a realizar transformações e ações para processar e extrair informações valiosas dos dados.

Prepare-se para mergulhar no universo do Apache Spark e descobrir como suas operações básicas podem ser aplicadas em uma variedade de cenários de análise de dados. Ao adquirir esse conhecimento, você estará capacitado(a) a enfrentar desafios complexos e a explorar todo o potencial do Spark para o processamento de grandes volumes de informações de maneira rápida, escalável e eficiente.

Não perca a oportunidade de dominar as operações básicas no Apache Spark e dar os primeiros passos em direção ao processamento de dados em escala. Com essa aula introdutória, você estará preparado(a) para expandir seus horizontes e embarcar em uma jornada de aprendizado contínuo em análise de Big Data.

HANDS ON

Vamos entender do que se tratam as transformações e ações no contexto do Apache Spark? Assista a aula para aplicarmos todos os conceitos de modo prático!

SAIBA MAIS

TRANSFORMAÇÕES E AÇÕES

No Apache Spark, as transformações e ações são operações fundamentais para processar e manipular dados distribuídos em RDDs (Resilient Distributed Datasets). Vamos explorar com mais detalhes as transformações e ações disponíveis no Spark.

Antes de apresentar os principais componentes de Transformações e Ações, é importante diferenciarmos o que é cada um.

Transformações

- As transformações no Spark são operações que criam um novo RDD a partir de um RDD existente.
- Elas são preguiçosas (lazy), o que significa que a execução real da transformação não ocorre imediatamente, apenas quando uma ação é chamada.
- As transformações são processadas de forma distribuída e podem ser executadas em paralelo em diferentes nós do cluster.
- As transformações são imutáveis, o que significa que elas não modificam o RDD original, mas criam um novo RDD com as modificações aplicadas.
- Exemplos de transformações incluem map, filter, reduceByKey, groupBy, join e muitas outras.

Ações

- As ações no Spark são operações que retornam resultados ou gravam dados.
- Elas desencadeiam a execução das transformações anteriores no RDD, iniciando o processamento real dos dados.
- As ações podem retornar resultados para o programa driver ou gravar dados em sistemas externos.
- Elas são executadas de forma distribuída, envolvendo os nós do cluster para processar os dados.

- Exemplos de ações incluem count, collect, take, reduce, saveAsTextFile e outras.

Exemplos de Transformações

- Map: aplica uma função a cada elemento do RDD e retorna um novo RDD com os resultados correspondentes.
- Filter: retorna um novo RDD contendo apenas os elementos que satisfazem uma determinada condição.
- FlatMap: é similar ao Map, porém cada elemento de entrada pode gerar zero ou mais elementos de saída. Os resultados são achatados em um único RDD.
- GroupBy: agrupa os elementos do RDD com base em uma determinada chave e retorna um novo RDD de pares (chave, lista de valores).
- ReduceByKey: agrega os valores de cada chave em um RDD, aplicando uma função de redução.
- SortBy: ordena os elementos do RDD com base em uma chave específica.
- Join: combina dois RDDs com base em chaves correspondentes, gerando um novo RDD contendo os pares combinados.
- Union: une dois RDDs em um único RDD, preservando todos os elementos.
- Distinct: retorna um novo RDD com elementos únicos, removendo duplicatas.

Exemplos de Ações

- Count: retorna o número de elementos no RDD.
- Collect: coleta todos os elementos do RDD e os retorna como uma lista no programa driver. Útil para RDDs pequenos, pois todos os dados são retornados para a máquina local.
- First: retorna o primeiro elemento do RDD.
- Take: retorna os primeiros elementos do RDD como uma lista.

- Reduce: aplica uma função de redução a todos os elementos do RDD e retorna o resultado final.
- Foreach: aplica uma função a cada elemento do RDD. É útil para executar operações de efeito colateral, como gravar em um banco de dados ou escrever em um sistema de arquivos.
- SaveAsTextFile: grava os elementos do RDD em um arquivo de texto no sistema de arquivos.
- CountByValue: conta o número de ocorrências de cada valor no RDD e retorna os resultados como um mapa (chave, contagem).
- Sum: retorna a soma de todos os elementos numéricos no RDD.

Exemplos práticos de transformações

1. Começaremos importando os módulos necessários:

```
from pyspark.sql import SparkSession
from pyspark.sql.functions import col
```

2. Agora, vamos criar uma sessão Spark:

```
spark = SparkSession.builder \
    .appName("Exemplos PySpark") \
    .getOrCreate()
```

```
# Carregar dados de um arquivo CSV
    df = spark.read.csv("/arquivo_teste.csv", header=True, inferSchema=True)

# Exibir o schema do DataFrame
    df.printSchema()
```

```
# Selecionar colunas específicas
    df_selected = df.select("coluna1", "coluna2")

# Filtrar registros com base em uma condição
    df_filtered = df.filter(col("coluna3") > 100)

# Adicionar uma nova coluna calculada
    df_with_new_column = df.withColumn("nova_coluna", col("coluna1") * 2)

# Renomear uma coluna
    df_renamed_column = df.withColumnRenamed("coluna1", "novo_nome_coluna")
```

Exemplos práticos de ações

1. Vamos começar importando os módulos necessários:

```
from pyspark.sql import SparkSession
from pyspark.sql.functions import col
```

2. Agora, crie uma sessão Spark:

```
spark = SparkSession.builder \
   .appName("Exemplos PySpark") \
   .getOrCreate()
```

```
# Carregar dados de um arquivo CSV
    df = spark.read.csv("/arquivo_teste.csv", header=True, inferSchema=True)
```

3. Exemplo 2: executando ações em um DataFrame.

```
# Contar o número total de registros
    count = df.count()
    print("Número total de registros: ", count)
```

```
# Exibir uma amostra dos dados
    df_sample = df.sample(fraction=0.1, seed=42)
    df_sample.show()

# Calcular a média de uma coluna
    average = df.select("coluna1").groupBy().avg().collect()[0][0]
    print("Média da coluna1: ", average)

# Salvar o DataFrame em um arquivo CSV
    df.write.csv("/arquivo_teste.csv", header=True)
```

O QUE VOCÊ VIU NESTA AULA?

Aprendemos que o Apache Spark é uma plataforma que oferece transformações e ações como operações fundamentais para processar e manipular dados distribuídos em RDDs (Resilient Distributed Datasets). As transformações são operações que criam um novo RDD a partir de um RDD existente.

Além disso, as transformações são processadas de forma distribuída, em paralelo, e são imutáveis, ou seja: não modificam o RDD original, mas criam um novo com as modificações aplicadas. Exemplos de transformações incluem map, filter, reduceByKey, groupBy e join.

As ações, por sua vez, são operações que retornam resultados ou gravam dados. Elas desencadeiam a execução das transformações anteriores no RDD, iniciando o processamento real dos dados. As ações podem retornar resultados para o programa driver ou gravar dados em sistemas externos. Assim como as transformações, as ações são executadas de forma distribuída, envolvendo os nós do cluster para processar os dados.

Exemplos de ações incluem count, collect, take, reduce e saveAsTextFile. Em resumo, as transformações e ações no Spark permitem realizar operações complexas de processamento e manipulação de dados distribuídos, aproveitando a capacidade de paralelismo e distribuição do ambiente de computação distribuída fornecido pelo Spark.

O que achou desta aula? Conte-nos no Discord! Estamos disponíveis na comunidade para te ajudar com dúvidas, comunicar avisos e muito mais. Os(as) seus(suas) colegas também estão por lá. Vamos?

REFERÊNCIAS

APACHE STARK. **Documentação Oficial Apache Spark**. Disponível em: https://spark.apache.org/documentation.html. Acesso em: 28 jun. 2023.

PENCHIKALA, S. **Big Data com Apache Spark Parte 1 – Introdução**. 2015. Disponível em: https://www.infoq.com/br/articles/apache-spark-introduction/. Acesso em: 28 jun. 2023.

RELVA, C. **Apache Spark**. 2015. Disponível em: https://www.ime.usp.br/~gold/cursos/2015/MAC5742/reports/ApacheSpark.pdf. Acesso em: 28 jun. 2023.

PALAVRAS-CHAVE

Palavras-chave: Apache Spark, Big Data, SQLContext, Hadoop, MapReduce, Big Data, Spark, Dados, Processamento.