GeekBrains

BigData. Введение в экосистему Hadoop.

Урок 5. Форматы хранения

На уроке изучаем популярные форматы хранения данных. Пробуем использовать различные форматы и сравнить их недостатки и преимущества.

Оглавление

Теоретическая часть

Строчные форматы

CSV - текстовый формат, предназначенный для представления табличных данных. Строка таблицы соответствует строке текста, которая содержит одно или несколько полей, разделенных запятыми.

JSON (англ. JavaScript Object Notation) — текстовый формат обмена данными, основанный на JavaScript. Как и многие другие текстовые форматы, JSON легко читается людьми. Формат JSON был разработан Дугласом Крокфордом. Несмотря на происхождение от JavaScript (точнее, от подмножества языка стандарта ECMA-262 1999 года), формат считается независимым от языка и может использоваться практически с любым языком программирования. Для многих языков существует готовый код для создания и обработки данных в формате JSON.

XML - это язык разметки подобный HTML. Расшифровывается как (англ. Extensible Markup Language - Расширяемый Язык Разметки) и является рекомендацией сообщества W3C в качестве языка разметки общего назначения (W3C recommended). В отличии от остальных языков разметки, XML сам по себе не определен (это означает, что вы должны сами определять используемые теги). Основной целью XML является передача данных между разными системами (даже концептуально разными), такими как интернет.

Колоночные форматы

Apache Parquet - это бинарный, колоночно-ориентированный (столбцовый) формат хранения данных, изначально созданный для экосистемы Hadoop. Apache Parquet был создан с целью сделать преимущества сжатого и эффективного столбцового представления данных доступными для любого проекта в экосистеме Hadoop. Он позволяет задавать схемы сжатия на уровне столбцов и рассчитан на возможность добавлять новые кодировки по мере их изобретения и реализации.

ORC (Optimized Row Columnar) – это колоночно-ориентированный (столбцовый) формат хранения Big Data в экосистеме Apache Hadoop. Он совместим с большинством сред обработки больших данных в среде Apache Hadoop и похож на другие колоночные форматы файлов: RCFile и Parquet. Формат ORC был разработан в феврале 2013 года корпорацией Hortonworks в сотрудничестве с Facebook, а месяц спустя Cloudera и Twitter представили Apache Parquet.

Бинарные форматы

Apache Avro — система сериализации данных, разработанная в рамках проекта Hadoop. Система использует JSON для определения структуры данных (схемы), которые сериализуются в компактный бинарный формат.

Практическая часть

Создадим простую таблицу в формате parquet:

```
create table lesson5.parquet_test (
    id int,
    str string)
stored as parquet;
```

Вставим значение и проверим таблицу:

```
insert into lesson5.parquet_test
select
1 as id,
"myText" as str;

select * from parquet_test;
```

Найдём созданные файлы в hdfs в папке /warehouse/tablespace

Попробуем изменить метаданные таблицы, и проверим, что сам файл не изменился:

```
alter table lesson5.parquet_test change str str_new string;
alter table lesson5.parquet_test change str_new str string;
```

Зададим параметр сжатия и попробуем создать другую таблицу

```
SET parquet.compression=SNAPPY;
```

Сравниваем таблицы с помощью parquet-tools mets и hdfs dfs -du

Создаем таблицу в формате avro:

```
create table lesson5.avro_test (
    id int,
    str string)
stored as avro;
```

```
insert into lesson5.avro_test
select
2 as id,
"myText" as str;
```

Устанавливаем сжатие и создаём ещё одну таблицу:

```
SET avro.output.codec=snappy;
```

Сравниваем параметры с помощьюу avro-tools

```
avro-tools getmeta
hdfs://manager.novalocal:8020/user/hive/warehouse/lesson5.db/avro_test/000000_0_copy_1
```

Создаем таблицу в формате ORC

```
create table lesson5_10.orc_test (
    id int,
    str string)
stored as orc;


insert into lesson5_10.orc_test
select
2 as id,
"myText" as str;
```

Проверяем с помощью команды --orcfiledump:

```
hive --orcfiledump /user/hive/warehouse/lesson5_10.db/orc_test/000000_0_copy_1
```

Домашнее задание

Нужно загрузить достаточно большой датасет и провести сравнительные эксперименты. Ниже примеры запросов

```
create external table hive_db.citation_data
(
        oci string,
        citing string,
        cited string,
        creation string,
        timespan string,
        journal_sc string,
        author_sc string
)
ROW FORMAT SERDE 'org.apache.hadoop.hive.serde2.OpenCSVSerde'
location '/test_datasets/citation'
```

Её размер можно узнать вот так :

```
hdfs dfs -du -h -s /test_datasets/citation
```

Что вам нужно сделать

1. Создать таблицы в форматах PARQUET/ORC/AVRO с компрессией и без. (Выберите несколько вариантов, например ORC с компрессией)
2. Заполнить данными из большой таблицы hive_db.citation_data
3. Посмотреть на получившийся размер данных
4. Посчитать count некоторых колонок в разных форматах хранения.
5. Посчитать агрегаты по одной и нескольким колонкам в разных форматах.
6. Сделать выводы о эффективности хранения и компрессии.

Задачи со * предназначены для продвинутых учеников, которым мало сделать обычное ДЗ.

Используемая литература

Для подготовки данного методического пособия были использованы следующие ресурсы:

1. https://ru.wikipedia.org/wiki/CSV
2. https://ru.wikipedia.org/wiki/JSON
3. https://ru.wikipedia.org/wiki/XML
4. https://developer.mozilla.org/ru/docs/Web/XML/XML_introduction
5. https://ru.bmstu.wiki/Apache_Parquet
6. https://ru.bmstu.wiki/Apache_Avro