BIGDATA TEAM

HW #08: Real Time

автор задания:

- Vybornov Artyom, avybornov@bigdatateam.org
- Big Data Instructor @ BigData Team
- Head of Data Platform @ Rambler Group

1. Описание задания

В данном ДЗ нужно решить 2 задачи. Решение надо выполнить с помощью Spark Structured Streaming.

WARNING: маловероятно, но при условии перезагрузки (или прочих проблем) на сервере типа client, поток данных в Kafka может быть прерван. Для возобновления потока данных обратитесь в чатике курса к преподавателям и/или поддержке курса. При отсутствии стрима свежих данных попробуйте установить отступы на чтение данных из Kafka вручную (подробнее - Structured Streaming Kafka Integration).

2. Критерии оценивания

Веса задач:

1. 50%
2. 50%

Балл за задачу складывается из:

- **80%** - правильное решение задачи
- **20%** - поддерживаемость и читаемость кода
  - в общем случае см. Clean Code и Google Python Style Guide
  - оценка качества будет проводиться автоматическим вызовом pylint:
    - `pylint *.py -d invalid-name,missing-docstring --persistent=n --ignored-modules=pyspark.sql.functions`
    - качество кода должно оцениваться выше 8.0 / 10.0
    - проверяем код **Python версии 3** с помощью `pylint==2.5.3`
- **0%** - эффективность решения (такие как потребляемые CPU-ресурсы, скорость выполнения (в предположении свободного кластера)).

Discounts (скидки и другие акции):

- **100%** за плагиат в решениях (всем участникам процесса)
- **100%** за посылку решения после deadline
- **5%** за каждую новую посылку (одна дополнительная посылка бесплатно)

Формула подсчета финальной оценки[1]:

$$max(0,\ 0.95^{max(0,\#\text{доп.посылок} - 1)} * (1 - \text{штраф. за. дедлайн. и. списывание})) * \text{последняя. оценка. из. } grader$$

[1] результат округляется до целого

3. Описание данных

- Входные данные - поток событий просмотра страниц в Kafka
- Брокеры кафка:
  `brain-node1.bigdatateam.org:9092,brain-node2.bigdatateam.org:9092,brain-node3.bigdatateam.org:9092`
- Топик кафка:
  `page_views`
- Формат строчки: `tsv`
- В каждой строке находятся следующие поля, разделенные знаком табуляции:
  - `DOUBLE` - TS (unixtime) события,
  - `STRING` - UID пользователя,
  - `STRING` - URL,
  - `STRING` - Title страницы,
  - `STRING` - User-Agent пользователя,

Пример:

```
1522588842.557  1129fa876d6a79497387723a77d3f24c
https://www.adamas.ru/catalog/kolca/?utm_medium=cpc&utm_source=yandex.direct&utm_campaign=Koltsa_Msk_RSYA%7c15392911&utm_term=%25D0%25BA%25D0%25BE%25D0%25BB%25D1%258C%25D1%2586%25D0%25BE&utm_content=k50id%7c01000000461487268 3_%7ccid%7c15392911%7cgid%7c1053311384%7caid%7c5569400968%7cadp%7cno%7cpos%7cnone0%7csrc%7ccontext_com.yandex.browser%7cdvc%7cmobile%7cmain&k50id=010000004614872683_&_openstat=ZGlyZWN0LnlhbmRleC5ydTsxNTM5MjkxMTs1NTY5NDAwOTY4O2NvbS55YW5kZXguYnJvd3NlcjpndWFyYW50ZWU&yclid=1620688752103923060
%D0%97%D0%BE%D0%BB%D0%BE%D1%82%D1%8B%D0%B5%20%D0%BA%D0%BE%D0%BB%D1%8C%D1%86%D0%B0%20-%20%D0%BA%D1%83%D0%BF%D0%B8%D1%82%D1%8C%20%D0%BA%D0%BE%D0%BB%D1%8C%D1%86%D0%BE%20%D0%B8%D0%B7%20%D0%B7%D0%BE%D0%BB%D0%BE%D1%82%D0%B0%20%D0%B2%20%D0%B8%D0%BD%D1%82%D0%B5%D1%80%D0%BD%D0%B5%D1%82-%D0%BC%D0%B0%D0%B3%D0%B0%D0%B7%D0%B8%D0%BD%D0%B5%20Adamas.ru      Mozilla/5.0 (Linux; Android 7.1.2; Redmi 5 Plus Build/N2G47H) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/63.0.3239.132 YaBrowser/18.1.1.645.00 Mobile Safari/537.36
1522588842.564  fe2042e800cbb63cff03f1152ebf74b6
https://www.gtavicecity.ru/gta-4/mods/
%D0%9C%D0%BE%D0%B4%D1%8B%20%D0%B4%D0%BB%D1%8F%20GTA%204%20%D1%81%20%D0%B0%D0%B2%D1%82%D0%BE%D0%BC%D0%B0%D1%82%D0%B8%D1%87%D0%B5%D1%81%D0%BA%D0%BE%D0%B9%20%D1%83%D1%81%D1%82%D0%B0%D0%BD%D0%BE%D0%B2%D0%BA%D0%BE%D0%B9%3A%20%D1%81%D0%BA%D0%B0%D1%87%D0%B0%D1%82%D1%8C%20%D0%B1%D0%B5%D1%81%D0%BF%D0%BB%D0%B0%D1%82%D0%BD%D0%BE%20%D0%BC%D0%BE%D0%B4%D1%8B%20%D0%B4
```

```
%D0%BB%D1%8F%20GTA%20IV      Mozilla/5.0 (Windows NT 6.2; WOW64)
AppleWebKit/537.36 (KHTML, like Gecko) Chrome/55.0.2883.87
UBrowser/7.0.185.1002 Safari/537.36
1522588842.564  dc215986678c3b4190a102db669cf86d
https://utro.ru/politics/2018/03/29/1355676.shtml?utm_campaign=utro&utm
_medium=referral&utm_source=push
%D0%9C%D0%BE%D1%81%D0%BA%D0%B2%D0%B0%20%D0%B6%D0%B5%D1%81%D1%82%D0%BA%D
0%BE%20%D0%BE%D1%82%D0%BF%D0%BB%D0%B0%D1%82%D0%B8%D0%BB%D0%B0%20%D0%A1%
D0%A8%D0%90%20%D0%B7%D0%B0%20%D1%81%D0%B2%D0%BE%D0%B8%D1%85%20%D0%B4%D0
%B8%D0%BF%D0%BB%D0%BE%D0%BC%D0%B0%D1%82%D0%BE%D0%B2%20%3A%3A%20%D0%9E%D
1%82%D1%80%D0%B0%D0%B2%D0%BB%D0%B5%D0%BD%D0%B8%D0%B5%20%D0%A1%D0%BA%D1%
80%D0%B8%D0%BF%D0%B0%D0%BB%D1%8F     Mozilla/5.0 (Linux; Android 7.0;
MI 5 Build/NRD90M) AppleWebKit/537.36 (KHTML, like Gecko)
Chrome/65.0.3325.109 Mobile Safari/537.36
```

4. Задача #1 (Task ID: realtime.domain_stat): статистика посещения доменов

В этом домашнем задании вам предстоит определить наиболее популярные домены по посещаемости и подсчитать число уников (то есть уникальных пользователей), которые зашли на этот домен.

Условия:

- Решение должно быть написано на Spark Structured Streaming.
- Ваше решение должно печатать в STDOUT топ-10 самых популярных (по просмотрам) доменов с информацией об общем числе просмотров этого домена и числа уников, которые на него зашли.
- Результат это кумулятивная статистика за всё время работы Streaming отсортированная по убыванию числа просмотров.
- Результат должен выводиться в консоль каждые 5 секунд
    - Если ваш код не успевает уложиться в этот интервал - возможно проблема в избыточном числе партиций
    - Важно выводить таблицу целиком и не обрезать длину столбцов (опция truncate должна быть выключена)

Пример результата:

```
-------------------------------------------
Batch: 10
-------------------------------------------
+---------------------------+----+------+
|domain                     |view|unique|
+---------------------------+----+------+
|news.rambler.ru            |18  |15    |
|m.lenta.ru                 |9   |7     |
|yandex.ru                  |9   |8     |
|www.championat.com         |7   |7     |
|www.yaplakal.com           |7   |7     |
|www.mk.ru                  |7   |7     |
|www.gazeta.ru              |6   |6     |
|www.coins-spb.ru           |6   |1     |
|miss-tramell.livejournal.com|6   |6     |
|woman.rambler.ru           |6   |5     |
+---------------------------+----+------+
```

5. Задача #2 (Task ID: realtime.runet_stat): оконная статистика посещения рунета

В этом домашнем задании вам предстоит определить видимый трафик в зоне ru и в остальном интернете. Сравнение производится на окне размером в 2 секунды каждую секунду (нас в обоих случаях интересует время события (поле TS из лога), а не обработки). Для трафика требуется подсчитать характеристики: число просмотров и число уников.

Условия:

- Решение должно быть написано на Spark Structured Streaming.
- Ваше решение должно печатать в STDOUT агрегированную статистику для сайтов зоны RU и остальных.
- Статистика это число просмотров и число уников которые в определенный интервал зашли на искомую группу доменов.
- Статистика рассчитывается за две секунды лога каждую секунду (под временем здесь подразумевается именно время события)
- Результат это кумулятивная статистика за всё время работы Streaming отсортированная по времени окна и убыванию числа просмотров в каждом окне.
- Решение должно выводить в консоль только первые 20 результатов работы

- Результат выводиться в консоль по мере готовности (:
  - Важно выводить таблицу целиком и не обрезать длину столбцов (опция truncate должна быть выключена)

Пример результата:

```
-------------------------------------------
Batch: 6
-------------------------------------------
+------------------------------------------+------+----+------+
|window                                    |zone  |view|unique|
+------------------------------------------+------+----+------+
|[2018-04-01 16:20:43, 2018-04-01 16:20:45]|ru    |719 |683   |
|[2018-04-01 16:20:43, 2018-04-01 16:20:45]|not ru|242 |255   |
|[2018-04-01 16:20:44, 2018-04-01 16:20:46]|ru    |719 |702   |
|[2018-04-01 16:20:44, 2018-04-01 16:20:46]|not ru|259 |255   |
...
|[2018-04-01 16:20:49, 2018-04-01 16:20:51]|ru    |717 |668   |
|[2018-04-01 16:20:49, 2018-04-01 16:20:51]|not ru|265 |257   |
+------------------------------------------+------+----+------+
```

6. Правила оформления задания

Оформление задания:

- Код задания (Short name): **HW8:RealTime**.
- Выполненное ДЗ запакуйте в архив **MADEBD2021Q1**<Surname>_<Name>_HW#**.zip**, пример -- **MADEBD2021Q1**_Dral_Alexey_**HW8**.zip. (Проверяйте отсутствие пробелов и невидимых символов после копирования имени отсюда.[2]) Если ваше решение лежит в папке my_solution_folder, то для создания архива hw.zip на Linux и Mac OS выполните команду[3]:
  - `zip -r hw.zip my_solution_folder/*`
- На Windows 7/8/10: необходимо выделить все содержимое директории my_solution_folder/ нажать правую кнопку мыши на одном из выделенных объектов, выбрать в открывшемся меню "Отправить >", затем "Сжатая ZIP-папка". Теперь можно переименовать архив.
- Решения заданий должны содержаться в одной папке.
- Решение должно предоставлять CLI интерфейс со следующими параметрами:
  - Общие настройки (должны быть заполнены все)
    - --topic-name - имя топика
    - --starting-offsets - отступ с которого скрипт начинает работать

[2] Онлайн инструмент для проверки: https://www.soscisurvey.de/tools/view-chars.php
[3] Флаг -r значит, что будет совершен рекурсивный обход по структуре директории

        - --kafka-brokers - координаты брокеров Kafka
    - Настройка триггера (должен быть заполнен один из двух)
        - --processing-time - микробатчевый триггер по времени (запускает триггер с заданной настройкой)
        - --once - триггер который запустит вычисление датасета лишь раз
    - Пример запуска решения

```
...runet_stat.py --topic-name page_views --starting-offsets latest
--processing-time "5 second" --kafka-brokers
brain-node1.bigdatateam.org:9092,brain-node2.bigdatateam.org:9092,brain-node3.bi
gdatateam.org:9092
```

    - Пример кода решения для инициализации нужных параметров:

```
import argparse
parser = argparse.ArgumentParser()
parser.add_argument("--kafka-brokers", required=True)
parser.add_argument("--topic-name", required=True)
parser.add_argument("--starting-offsets", default='latest')

group = parser.add_mutually_exclusive_group()
group.add_argument("--processing-time", default='0 seconds')
group.add_argument("--once", action='store_true')

args = parser.parse_args()
if args.once:
    args.processing_time = None
else:
    args.once=None
...

   .trigger(once=args.once, processingTime=args.processing_time) \
...
```

- PySpark-скрипты для запуска решений следует называть `task_<Surname>_<Name>_<#task_ID.suffix>.py`:
    - решение задачи #1 должно называться "task_*_domain_stat.py" и его можно запустить с помощью команды:
        - `PYSPARK_DRIVER_PYTHON=python3.6 PYSPARK_PYTHON=python3.6 spark-submit --packages org.apache.spark:spark-sql-kafka-0-10_2.11:2.4.0 "task_*_domain_stat.py %cli_args%"`
    - решение задачи #2 должно называться "task_*_runet_stat.py" и его можно запустить с помощью команды:
        - `PYSPARK_DRIVER_PYTHON=python3.6 PYSPARK_PYTHON=python3.6 spark-submit --packages org.apache.spark:spark-sql-kafka-0-10_2.11:2.4.0 "task_*_runet_stat.py %cli_args%"`
    - скрипты выводят на экран (STDOUT) указанное в задании число строк в нужном формате каждый батч

- Вывод STDOUT задач с результатом обработки 4 батчей нужно сохранить в соответствующих файлах в архиве посылки домашнего задания (например, `task_*_suffix.out`).[4]
  - Формат файла произвольный
  - Этот файл не влияет на успешность сдачи грейдеру
- Перед проверкой убедитесь, что дерево вашего архива выглядит так:
  - `| MADEBD2021Q1_<Surname>_<Name>_HW8.zip`
  - `| ---- task_<Surname>_<Name>_domain_stat.py`
  - `| ---- task_<Surname>_<Name>_domain_stat.out`
  - `| ---- task_<Surname>_<Name>_runet_stat.py`
  - `| ---- task_<Surname>_<Name>_runet_stat.out`
  - При несовпадении дерева вашего архива с представленным деревом, ваше решение будет невозможно автоматически проверить, а значит, и оценить его.
- Для того, чтобы сдать задание необходимо:
  - Зарегистрироваться и залогиниться в сервисе Everest
  - Перейти на страницу приложения BDT-grader-MADE-BD
  - Выбрать вкладку Submit Job (если отображается иная).
  - Выбрать в качестве “Task” значение: **HW8:RealTime**[5]
  - Загрузить в качестве “Task solution” файл с решением
  - В качестве Sender ID указать тот, который был выслан по почте
- Если Вы видите надпись "You are not allowed to run this application" во вкладке Submit Job в Everest, то на данный момент сдача закрыта (нет доступных для сдачи домашних заданий, по техническим причинам или другое). Попробуйте, пожалуйста, еще раз через некоторое время. Если Вы еще ни разу не сдавали, у коллег сдача работает, но Вы видите такое сообщение, сообщите нам об этом.
- Ситуации:
  * система оценивания показывает оценку (Grade) < 0, а отчет (Grading report) не помогает решить проблему (пример помощи: в случае неправильно указанного Sender ID система вернет -2 и информацию о том, что его нужно поправить);
  * показывает 0 и в отчете (Grading report) не указано, какие тесты не пройдены.
  Если Вы столкнулись с какой-то из них присылайте ссылку на выполненное задание (Job) на почту с темой письма “Short name. ФИО.”. Например: “**HW8:RealTime**. Иванов Иван Иванович.”
  Пример ссылки: https://everest.distcomp.org/jobs/67893456230000abc0123def
  Внимание: Если до дедлайна остается меньше суток, и Вы знаете (сами проверили или коллеги сообщили), что сдача решений сломана, обязательно сдайте свое решение и напишите письмо, как написано выше, чтобы мы видели, какое решение Вы имели до дедлайна и смогли его оценить.

[4] Для подготовки архива с решением и выводом результатов запуска можно воспользоваться командой “tee”
[5] Сервисный ID: realtime.onsite_hw

- Перед отправкой задания, оставьте, пожалуйста, отзыв о домашнем задании по сссылке: http://rebrand.ly/mailbd2021q1_feedback_hw. Это позволит нам скорректировать учебную нагрузку по следующим заданиям (в зависимости от того, сколько часов уходит на решение ДЗ), а также ответить на интересующие вопросы.

Любые вопросы / комментарии / предложения можно писать в телеграм-канал курса или на почту bigdata_made2021q1@bigdatateam.org .

Всем удачи!