ПРОЕКТ 3
АНАЛИЗ ЛОГОВ

Климантов Артём (Lapis)

ЦЕЛИ ПРОЕКТА

1) Суррогатный ключ устройства
2) Название устройства
3) Количество пользователей
4) Доля пользователей данного устройства от общего числа пользователей
5) Количество совершенных действий для данного устройства
6) Доля совершенных действий с данного устройства относительно других устройств
7) Список из 5 самых популярных браузеров, используемых на данном устройстве различными пользователями, с указанием доли использования для данного браузера относительно остальных браузеров
8) Количество ответов сервера, отличных от 200 на данном устройстве
9) Для каждого из ответов сервера, отличных от 200, сформировать поле, в котором будет содержаться количество ответов данного типа

ОСНОВНЫЕ ПРОБЛЕМЫ ПРОЕКТА

- Каждая запись в файле хоть и содержит не так много информации, но из-за большого количества записей, а именно 10365152 различных строк, анализ файла как минимум не возможен на слабых кластерах, а как максимум занимает около 40 минут, что не является хорошим показателем если хочется проверять гипотезы.
- Для приемлемых сроков создания и отладки основного скрипта для парсинга пришлось «отрезать» небольшой кусок в 5к строк и проверять работу на нём, так как мой компьютер хоть и не является слабым, но в силу некоторых железных ограничений выполнить преобразования на полном файле не представляется возможным

ПЛАН РЕАЛИЗАЦИИ ПРОЕКТА

1. Распарсить текстовые строки на нужные данные с использованием регулярных выражений и парсинга через методы библиотеки httpagentparser
2. Вычислить необходимые количественные метрики
3. Создать и заполнить витрину данных в соответствии с запросами задачи

СТЕК ТЕХНОЛОГИЙ

- В силу объемности изначального журнала данных в виде обработчика был выбран Spark и его имплементация для питона PySpark.
- Анализ и расчёт всех необходимых метрик проводился с использованием Jupyter и библиотек Pandas и PySpark.
- Парсинг строк производился с использованием регулярных выражений и библиотеки httpagentparser
- Для передачи информации в базу данных использовался psycopg2.
- Для хранения витрины была создана таблица в PostgresSQL.

РЕЗУЛЬТАТ

- Jupyter Notebook содержащий необходимые ячейки после запуска которых в базе данных будет создан и заполнена таблица с метриками.
- Файл с данными витрины df_mart.csv
- Витрина данных Log_mart

Название	#	Тип данных
id	1	serial4
platform	2	text
device_users	3	int4
part_device_...	4	float8
device_actio...	5	int4
part_device_...	6	float8
browser_cnt	7	int4
part_browser	8	float8
answers_200	9	numeric
answers_ne...	10	numeric
answers_3xx	11	numeric
answers_4xx	12	numeric
answers_5xx	13	numeric

id	platform	device_users	part_device_users	device_actions	part_device_actions	browser_cnt
1	Android 4.2.1	1 944	0,01	1 944	0,01	1 944
2	Mac OS X 10.9	12	0	12	0	12
3	Windows 10	315 455	0,84	315 455	0,84	315 455
4	Windows 8	733 530	1,96	733 530	1,96	733 530
5	Mac OS X 10.7.3	745 200	1,99	745 200	1,99	745 200
6	Android 4.2.2	50 856	0,14	50 856	0,14	50 856
7	iOS 11.2.5	441	0	441	0	441
8	Android 4.0.4	324	0	324	0	324
9	Mac OS X 10.13.6	8	0	8	0	8
10	iOS 7.0	129 109	0,34	129 109	0,34	129 109
11	Android 4.4.4	16 783	0,04	16 783	0,04	16 783
12	Android 5.0.1	3 369	0,01	3 369	0,01	3 369
13	iOS 11.2.2	1	0	1	0	1
14	" like Gecko) Version/4.0 Chrome/69.0.3497.100 Mobile Safari/537.36 GSA/8	1	0	1	0	1
15	Windows 8.1	144 079	0,38	144 079	0,38	144 079
16	" like Gecko) Chrome/71.0.3578.99 Mobile Safari/537.36""""""""	3	0	3	0	3
17	Mac OS X 10.12.6	16	0	16	0	16
18	Mac OS X 10.10.2	4	0	4	0	4
19	Windows	1	0	1	0	1
20	Android 4.1.2 Persian Language By Forum.gpgstore.ir Team	4	0	4	0	4
21	Mac OS X 10.10.1) AppleWebKit/600.2.5 (KHTML, like Gecko) Version/8.0.2 S	332 930	0,89	332 930	0,89	332 930
22	iOS 11.0.3	49	0	49	0	49
23	Windows 10.0	1 469	0	1 469	0	1 469
24	" like Gecko) Chrome/70.0.3538.102 Safari/537.36 OPR/57.0.3098.116""""""""	302	0	302	0	302
25	Android 6.0	70 135	0,19	70 135	0,19	70 135
26	iOS 10.2	484	0	484	0	484
27	Android 7.0 SHA	6 561	0,02	6 561	0,02	6 561
28	iOS 11.4.1	310	0	310	0	310
29	Android 5.0	19 433	0,05	19 433	0,05	19 433
30	iOS 12.1.2	65 629	0,18	65 629	0,18	65 629
31	Android 4.3	14 830	0,04	14 830	0,04	14 830
32	Windows Vista	1 717 664	4,59	1 717 664	4,59	1 717 664
33	Windows 95	25	0	25	0	25
34	Android 7.1.1	24 527	0,07	24 527	0,07	24 527
35	Android 6.0.1	26 262 152	70,12	26 262 152	70,12	26 262 152
36	Mac OS X 10.14.2	4	0	4	0	4
37	" like Gecko) Version/4.0 Chrome/61.0.3163.98 Mobile Safari/537.36 GSA/7.'	80	0	80	0	80
38	" like Gecko) Chrome/70.0.3538.80 Mobile Safari/537.36""""""""	1	0	1	0	1
39	" like Gecko) Chrome/50.0.2661.89 Mobile Safari/537.36""""""""	1	0	1	0	1
40	Windows NT 9.0	16	0	16	0	16

ВЫВОДЫ В ХОДЕ ВЫПОЛНЕНИЯ

1) Самое важное это верно оценить мощность своего компьютера как с точки зрения таких комплектующих как оперативная память и количество ядер процессора, так и с точки зрения свободного места для создания кэша и прочего.
2) Jupyter очень хитрый и если переменная нигде не используется то до момента востребованности он ее даже трогать не станет, данная особенность сыграла со мной очень злую шутку, которая закончилась очень резким и неожиданным синим экраном смерти винды.
3) Show() – очень странный метод, который может «выплюнуть» 100 строк за секунду, а может над тремя думать пять минут
4) Если есть возможность, то обязательно сохранять промежуточные данные, ибо считать 20 минут каждый раз очень больно и долго.
5) Чем меньше лишнего тем лучше, не стоит плодить ворох переменных на любую свою идею, все лучше тестировать в изолированной среде.