

## HW #05: Spark RDD

---

1. Описание задания	<b>2</b>
2. Критерии оценивания	<b>2</b>
3. Описание данных	<b>2</b>
4.1 Задача #1 (Task ID: spark.bigram): народные биграммы	<b>3</b>
4.2 Задача #2 (Task ID: spark.collocation): коллокации	<b>4</b>
5. Правила оформления задания	<b>6</b>

---

автор задания: BigData Team, коллективная работа.



## 1. Описание задания

В данном ДЗ нужно решить **2 задачи**. Задачи общие для всех. Решение надо выполнить с помощью Apache Spark, можно использовать только RDD API.

## 2. Критерии оценивания

Веса задач:

1. 40%
2. 60%

Балл за задачу складывается из:

- **60%** - 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 --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`
- **20%** - 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 дополнительную посылку в тестирующую систему (одна дополнительная посылка бесплатно)

Формула подсчета финальной оценки<sup>1</sup>:

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

## 3. Описание данных

### 3.1 Дамп Википедии

en\_articles\_part:

---

<sup>1</sup> результат умножается на 10 (максимальная оценка) и округляется до первой цифры после точки



- Путь на кластере: полный<sup>2</sup> датасет - `/data/wiki/en_articles_part`
- Формат: текст
- В каждой строке находятся следующие поля, разделенные знаком табуляции:
  1. INT - id статьи,
  2. STRING - текст статьи,

*Пример:*

```
12      Anarchism      Anarchism is often defined as a political
philosophy which holds the state to be undesirable, unnecessary, or
harmful.
```

### 3.2 Стоп-слова

`stop_words_en:`

- Путь на кластере: `/data/stop_words/stop_words_en-хр06.txt`
- Формат: одно стоп-слово на строку

*Пример:*

```
...
wherein
whereupon
wherever
...
```

## 4.1 Задача #1 (Task ID: spark.bigram): народные биграммы

Найдите все пары двух последовательных слов (биграмм), где первое слово:

```
narodnaya
```

Для каждой пары подсчитайте количество вхождений в тексте статей Википедии. Выведите все пары с их частотой вхождений в лексикографическом порядке. Формат вывода:

```
word_pair <tab> count
```

Условия:

- для однозначности вычислений, выделите слова из статьи с помощью регулярного выражения `re.findall(r"\w+", text)`;
- привести все слова к нижнему регистру;

---

<sup>2</sup> Да, здесь нет ошибки, работаем на части данных, чтобы побыстрее познакомиться со Spark RDD

- слова в паре объединить символом нижнего подчеркивания “\_”;
- отсортировать слова в выводе по алфавиту;
- решение должно обрабатывать в течение 3х минут на свободном кластере (3 ноды x 8 CPU x 16GB RAM).

*Пример вывода:*

```
...  
crazy_zoo 42  
red_apple 100500  
...
```

## 4.2 Задача #2 (Task ID: spark.collocation): коллокации

Коллокация - это комбинации слов, которые часто встречаются вместе. Например, «high school» или «Roman Empire». Чтобы определить, является ли пара слов коллокацией, можно воспользоваться метрикой NPMI - нормализованная точечная взаимная информация.

Чтобы рассчитать NPMI, введем несколько определений:

1.  $P(a)$  - вероятность увидеть слово “a” в датасете.  
$$P(a) = \text{num\_of\_occurrences\_of\_word\_}a / \text{total\_number\_of\_words}$$
$$\text{total\_number\_of\_words}$$
 - общее количество слов в тексте
2.  $P(ab)$  - вероятность увидеть пару слов “a” и “b”, идущих подряд.  
$$P(ab) = \text{num\_of\_occurrences\_of\_pair\_}ab / \text{total\_number\_of\_word\_pairs}$$
$$\text{total\_number\_of\_word\_pairs}$$
 - общее количество пар
3.  $PMI(a,b) = \ln( P(ab) / [P(a) \times P(b)] )$
4.  $NPMI(a,b) = PMI(a,b) / -\ln(P(ab))$  - величина PMI нормализованная в диапазон  $[-1, 1]$ ;

Примеры и комментарии:

- значение NPMI равное “-1” будет означать, что пара слов никогда не встречается в датасете. Например, такие пары как **“green idea”** или **“sleeps furiously”** никогда не встречаются вместе, поэтому  $P(ab) = 0$ , следовательно  $PMI(a,b) = -\inf$ ,  $NPMI = -1$ ;
- значение NPMI равное “0” будет означать, что слова в паре встречаются абсолютно независимо друг от друга. Рассмотрим пример **“the doors”**: “the” может встретиться рядом с любым словом. Таким образом,  $P(ab) = P(a) \times P(b)$  и  $PMI(a,b) = \ln(1) = 0$ ,  $NPMI = 0$ ;

- значение NPMI равное "1" будет означать, что это идеальная коллокация. Предположим, что "Roman Empire" - это уникальная комбинация, и за каждым появлением "Roman" следует "Empire", и, наоборот, каждому появлению "Empire" предшествует "Roman". В этом случае  $P(ab) = P(a) = P(b)$ , поэтому  $PMI(a,b) = -\ln(P(a)) = -\ln(P(b))$ , следовательно  $NPMI = 1$ .

#### Условия:

- найти самые популярные коллокации в Википедии;
- для однозначности вычислений, выделите слова из статьи с помощью регулярного выражения `re.findall(r"\w+", text)`;
- привести все слова к нижнему регистру;
- удалить стоп-слова;
- слова в паре объединить символом нижнего подчеркивания "\_";
- отфильтровать биграммы, которые встретились **не реже 500 раз** (т.е. проводим все необходимые join'ы и считаем NPMI только для них, НО оценку вероятности встретить бигramму, считаем на полном датасете);
- отсортировать слова в выводе по значению NPMI;
- вывести **ТОП-39** самых популярных коллокаций и их значения NPMI (округляем до 3-го знака после запятой, см. round);
- решение должно отрабатывать в течение 3х минут на свободном кластере (3 ноды x 8 CPU x 16GB RAM).

#### Формат вывода:

```
word_pair <tab> npmi
```

#### Пример вывода:

```
...
south_africa      0.619
roman_empire      0.603
...
```

## 5. Правила оформления задания

Оформление задания:

- Код задания (Short name): **HW5:Spark-RDD**.
- Выполненное ДЗ запакуйте в архив `MADEBD2021Q1_<Surname>_<Name>_HW#.zip`, например, для Алексея Драля - `MADEBD2021Q1_Dral_Alexey_HW5.zip`. Если ваше решение лежит в папке `my_solution_folder`, то для создания архива `hw.zip` на Linux и Mac OS выполните команду<sup>3</sup>:
  - `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 должно содержаться в одной папке.
- PySpark-скрипты для запуска решений следует называть `task_<Surname>_<Name>_<#task_ID>.py`:
  - решение задачи #1 должно называться `"task_*_bigram.py"` и его можно запустить с помощью команды:
    - `PYSPARK_DRIVER_PYTHON=python3.6 PYSPARK_PYTHON=python3.6 spark-submit "task_*_bigram.py"`
  - решение задачи #2 должно называться `"task_*_collocation.py"` и его можно запустить с помощью команды:
    - `PYSPARK_DRIVER_PYTHON=python3.6 PYSPARK_PYTHON=python3.6 spark-submit "task_*_collocation.py"`
  - скрипты выводят на экран (STDOUT) указанное в задании число строк в нужном формате
- **Вывод STDOUT задач нужно сохранить в соответствующих файлах в архиве посылке домашнего задания (например, `task_*_bigram.out`).<sup>4</sup>**
- Перед проверкой убедитесь, что дерево вашего архива выглядит так:
  - | `MADEBD2021Q1_<Surname>_<Name>_HW5.zip`
  - | `---- task_<Surname>_<Name>_bigram.py`
  - | `---- task_<Surname>_<Name>_bigram.out`
  - | `---- task_<Surname>_<Name>_collocation.py`
  - | `---- task_<Surname>_<Name>_collocation.out`
  - При несовпадении дерева вашего архива с представленным деревом, ваше решение будет невозможно автоматически проверить, а значит, и оценить его.

<sup>3</sup> Флаг `-r` значит, что будет совершен рекурсивный обход по структуре директории

<sup>4</sup> Для подготовки архива с решением и выводом результатов запуска можно воспользоваться командой `"tee"`



- 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: **HW5:Spark-RDD<sup>5</sup>**
  - 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: **"HW5:Spark-RDD. 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.
- Перед отправкой задания, оставьте, пожалуйста, отзыв о нём по ссылке: [http://rebrand.ly/mailbd2021q1\\_feedback\\_hw](http://rebrand.ly/mailbd2021q1_feedback_hw). Это позволит 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 [Discord-канал курса](#) или на почту [bigdata\\_made2021q1@bigdatateam.org](mailto:bigdata_made2021q1@bigdatateam.org).

Всем удачи!

---

<sup>5</sup> Сервисный ID: spark.rdd.onsite\_hw