Schemat i transformacja danych

22 października 2018

Opis pliku z zadaniami

Wszystkie zadania na zajęciach będą przekazywane w postaci plików `.pdf`, sformatowanych podobnie do tego dokumentu. Zadania będą różnego rodzaju. Za każdym razem będą one odpowiednio oznaczone:

- Zadania do wykonania na zajęciach oznaczone są symbolem $\triangle$ – nie są one punktowane, ale należy je wykonać w czasie zajęć.
- Punktowane zadania do wykonania na zajęciach oznaczone są symbolem $\diamond$ – należy je wykonać na zajęciach i zaprezentować prowadzącemu, w wypadku nie wykonania zadania w czasie zajęć lub nieobecności, zadania staje się zadaniem do wykonania w domu ($\star$).
- Zadania do wykonania w domu oznaczone są symbolem $\star$ – są one punktowane, należy je dostarczyć w sposób podany przez prowadzącego i w wyznaczonym terminie (zwykle przed kolejnymi zajęciami).
- Zadania programistyczne można wykonywać w dowolnym języku programowania, używając jedynie biblioteki standardowej dostępnej dla tego języka.

1 Studium przypadku △

Treść

Podczas zajęć laboratoryjnych będziemy używać danych związanych z konkursem *Million Song Dataset Challenge*. Dotyczy on stworzenia systemu rekomendującego piosenki dla użytkowników pewnego serwisu. Dokładny opis danych można znaleźć na stronach:

- http://www.kaggle.com/c/msdchallenge
- http://labrosa.ee.columbia.edu/millionsong/

W naszych zadaniach zrobimy pierwsze kroki w kierunku stworzenia systemu rekomendacyjnego.

Dodatkowo podczas zajęć zapoznamy się z techonologią Docker (https://www.docker.com/), która posłuży nam do opakowania opracowanych rozwiązań.

2 Przetwarzanie dużych danych *10p.* ◇

Treść

Na zajęciach będziemy wykorzystywać okrojony i zmodyfikowany zbiór danych Million Song Dataset (MSD). Należy pobrać dwa pliki ze strony przedmiotu z następującymi informacjami:

- `unique_tracks.txt` — zawiera informacje takie jak identyfikator utworu, identyfikator wykonania, nazwę artysty oraz tytuł utworu,
- `triplets_sample_20p.txt` — zawiera identyfikator użytkownika, identyfikator utworu oraz datę odsłuchania.

Przygotuj `Dockerfile` w którym przy pomocy dowolnej techonologi programistycznej/bazodanowej otrzymasz informacje na temat:

1. Ranking popularności piosenek,
2. Ranking użytkowników ze względu na największą liczbę odsłuchanych unikalnych piosenek,
3. Artysta z największą liczbą odsłuchań,
4. Sumaryczna liczba odsłuchań w podziale na poszczególne miesiące,
5. Wszyscy użytkownicy, którzy odsłuchali wszystkie trzy najbardziej popularne piosenki zespołu Queen.

W celu przygotowania `Dockerfile` zapoznaj się z:

1. https://www.docker.com/
2. https://docs.docker.com/get-started/
3. https://docs.docker.com/get-started/part2/

Oraz przykładem dostępnym na stronie przedmiotu.
Rozwiązanie powinno działać zgodnie z poniższą specyfikacją techniczną:

1. Załóż, że pliki `unique_tracks.txt` oraz `triplets_sample_20p.txt` znajdują się w katalogu razem z Twoim `Dockerfile` i możesz skopiować je do wnętrza swojego obrazu.
2. Dla podpunktu 1. wypisz na standardowe wyjście 10 najpopularniejszych piosenek posortowanych w kolejności od najpopularniejszej w formacie:

```
<tytuł piosenki 1> <nazwa wykonawcy piosenki 1> <ilość odsłuchań piosenki 1>
<tytuł piosenki 2> <nazwa wykonawcy piosenki 2> <ilość odsłuchań piosenki 2>
...
<tytuł piosenki 10> <nazwa wykonawcy piosenki 10>  <ilość odsłuchań piosenki 10>
```

3. Dla podpunktu 2. wypisz na standardowe wyjście 5 najpopularniejszych piosenek w formacie:

```
<id użytkownika 1> <ilość odsłuchanych unikatowych piosenek przez użytkownika 1>
<id użytkownika 2> <ilość odsłuchanych unikatowych piosenek przez użytkownika 2>
...
<id użytkownika 10> <ilość odsłuchanych unikatowych piosenek przez użytkownika 10>
```

4. Dla podpunktu 3. wypisz na standardowe wyjście nazwę najpopularniejszego wykonawcy w formacie:

```
<nazwa wykonawcy> <sumaryczna ilość odsłuchań jego piosenek>
```

5. Dla podpunktu 4. wypisz na standardowe wyjście numery miesiące w formacie:

```
1 <sumaryczna ilość odsłuchań w miesiącu 1>
2 <sumaryczna ilość odsłuchań w miesiącu 2>
3 <sumaryczna ilość odsłuchań w miesiącu 3>
...
12 <sumaryczna ilość odsłuchań w miesiącu 12>
```

   lub

```
01 <sumaryczna ilość odsłuchań w miesiącu 1>
02 <sumaryczna ilość odsłuchań w miesiącu 2>
03 <sumaryczna ilość odsłuchań w miesiącu 3>
...
12 <sumaryczna ilość odsłuchań w miesiącu 12>
```

6. Dla podpunktu 5. wypisz na standardowe wyjście 10 pierwszy id użytkowników spełniających warunek zgodnie z porządkiem alfabetycznym w formacie:

```
<id użytkownika 1>
<id użytkownika 2>
...
<id użytkownika 10>
```

7. Obraz powinien wykonać zadanie po uruchomieniu bez potrzeby podawania żadnych dodatkowych parametrów (zawierać polecenie CMD/ENTRYPOINT).

8. Zadbaj o efektywny czas przetwarzania – całość nie powinna zająć więcej niż 20 min.

9. Do projektu dołącz plik README w którym opiszesz wybraną technologię i uzasadnienie swojego wyboru.

10. Swoje rozwiązanie wyślij w archiwum zip, zawierającym folder o nazwie: `<numer indeksu>_<imię>_<nazwisko>` na adres mwydmuch@cs.put.poznan.pl (tytule maila umieść najpierw prefiks [PMD]) Bezpośrednio wewnątrz tego folderu powinien znajdować się plik `Dockerfile`.

11. Twoje rozwiązanie zostanie sprawdzone sprawdzarką, dlatego postaraj się by Twoje rozwiązanie było zgodne ze specyfikacją!