04-rzepinskip-to_read_cleaning

January 4, 2019

1 Important

`make data` has to be run before any cell in this notebook

2 Imports

```
In [1]: import pandas as pd
```

```
In [2]: to_read_df = pd.read_csv("../data/raw/to_read.csv")
        ratings_df = pd.read_csv("../data/raw/ratings.csv")
```

We should use `ratings-train.csv`(training set of 90% of ratings) normally, but for reproducibility reasons we use full set.

3 Analysis

```
In [3]: merged_df = to_read_df.merge(ratings_df, on=['user_id', 'book_id'],
                                     how='left')
```

```
In [4]: len(merged_df[merged_df['rating'].notna()]) / len(to_read_df)
```

```
Out[4]: 0.0024104173856832165
```

We remove about 0.02% of data, so it has marginal impact.

```
In [5]: len(merged_df[merged_df['rating'].notna()])
```

```
Out[5]: 2200
```

```
In [6]: len(merged_df[merged_df['rating'].isnull()])
```

```
Out[6]: 910505
```

```
In [7]: merged_df.drop('rating', axis=1)
```

```
Out[7]:         user_id  book_id
        0             9        8
        1            15      398
        2            15      275
        3            37     7173
```

```
4           34     380
5           34     483
6           34    8598
7           34    3581
8           70     498
9           76    4250
10          94    1167
11          29    3508
12          29    4475
13          29     323
14          29     131
15          29    2304
16         105     233
17         113    6756
18         113    7127
19          29    2284
20          29     662
21         116     474
22         116    8697
23         124     682
24         124       5
25          94    4475
26          94    5704
27          94    1847
28         137     362
29          94    1239
...        ...     ...
912675   41259     852
912676   23042     146
912677   52948    6152
912678   52948    6814
912679   28938    9595
912680   50277    1693
912681   50277    8568
912682   10622    4589
912683   21682    3541
912684   53358     195
912685   53358    1065
912686   53358    1028
912687   53358    6107
912688   15447     235
912689   15447    6868
912690   36869    7844
912691    5237    2378
912692   45911    8362
912693   43806    1816
912694   45870     744
912695   45870    1499
```

```
912696    10622     2367
912697    42071     1952
912698    42071     7272
912699     7893      793
912700    39374     1049
912701    10492     5180
912702    21879     4827
912703    21879     6642
912704    48192     7773

[912705 rows x 2 columns]
```