

 Research Prediction Competition

Data Mining Hackathon on BIG DATA (7GB) Best Buy mobile web site

\$1,000

Predict which BestBuy product a mobile web visitor will be most interested in based on their search query or behavior over 2 years (7 GB). Prize Money

24 teams · 10 years ago

Data Description

The main data for this competition is in the `train.csv` and `test.csv` files. These files contain information on what items users clicked on after making a search.

Each line of `train.csv` describes a user's click on a single item. It contains the following fields:

- user: A user ID
- sku: The stock-keeping unit (item) that the user clicked on
- category: The category the sku belongs to
- query: The search terms that the user entered
- click_time: Time the sku was clicked on
- query_time: Time the query was run

`test.csv` contains all of the same fields as `train.csv` except for sku. It is your job to estimate which sku's were clicked on in these test queries.

Due to the internal structure of BestBuy's databases, there is no guarantee that the user clicks resulted from a search with the given query. What we do know is that the user made a query at `query_time`, and then, at `click_time`, they clicked on the sku, but we don't know that the click came from the search results. The `click_time` is never more than five minutes after the `query_time`.

In addition, there is information about products, product categories, and product reviews in `product_data.tar.gz`.

We have also provided a sample benchmark submission and the code that produces it. `popular_skus.py` is a simple python script that finds the most popular skus in each product category, and then estimates that a user clicked on one of the five most popular skus in their product category. This script produces the benchmark in `popular_skus.csv`.

The syntax of a submission should be the same as that in `popular_skus.csv`: A file with the header "sku", and each of the following lines containing the space-delimited estimates of the clicked sku that resulted after the queries in `test.csv`, and in the same order.

Product Data Dictionary

<https://bbyopen.com/documentation/products-api/product-attributes#TableProdRefInfo>

For more BestBuy Data and APIs, check out <https://bbyopen.com/>






and more BigDataR tools https://github.com/koooeo/BigDataR_Examples/tree/master/ACM_comp

```
> kaggle competitions download -c acm-sf-chapter-hackathon-big
```





Data Explorer

1.25 GiB

 `popular_skus.csv`
 `popular_skus.py`
 `product_data.tar.gz`
 `test.csv`
 `train.csv`

Summary

▸  5 files
▸  12 columns

 Download All

< popular_skus.csv (50.63 MiB)



Detail Compact Column

1 of 1 columns ▾

sku
2550164 3247045 ... 10%
2620821 2138389 2... 4%
Other (1070853) 86%
1854151 2416092
9830614 2262047
2345191
8116585 1166875
9405128 9530144
6166301
2506173 2506119
1517163 2506146
2506094
9257163 5431927

8525651 3346391 1388362
2550164 3247045 3209231 3103247 3142315
2550164 3247045 3209231 3103247 3142315
9755322 1535836 3108109 9755395 1534115
8939839 8280834 9119642 8280843 8850601
3217488 3584076 9728756 8599313 3301653
3440086 9274714 7997055 8461149 2940973