TARGET
DATA SCIENCE

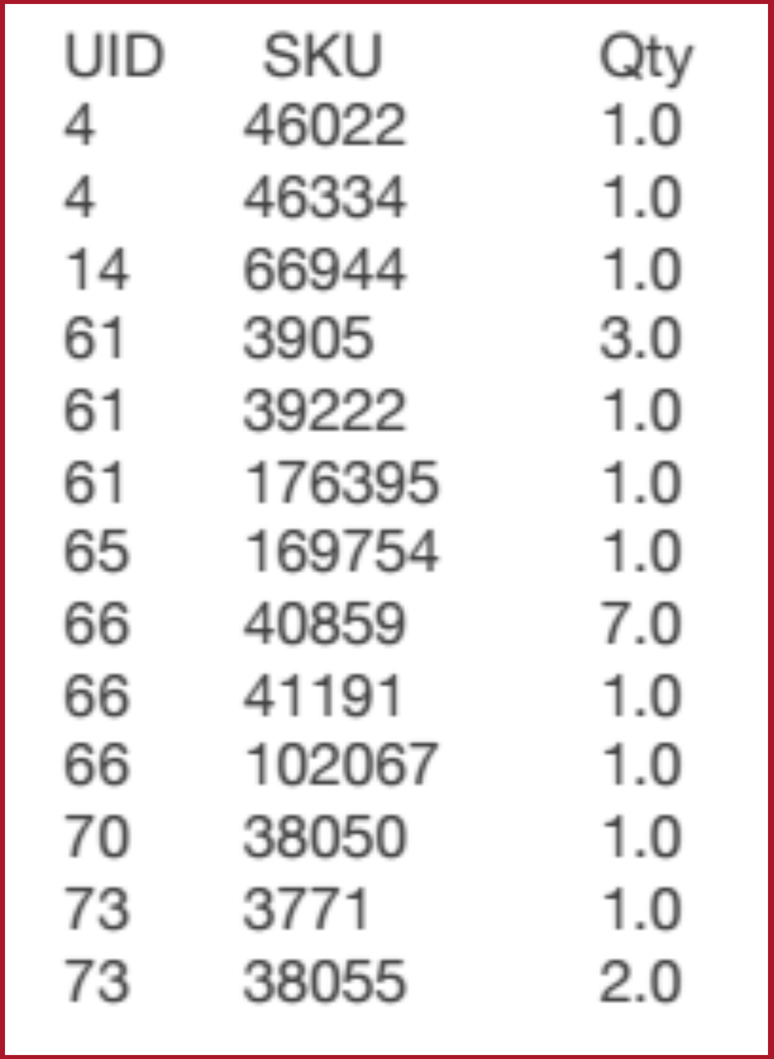
BUILD A PERSONALIZED RECOMMENDATION SYSTEM!

Given the user, product, qty and product, product text description data discover product taxonomy and jointly generate personalized recommendations for each user. For each user, provide top 5 recommendations in a JSON format.

OUTPUT EXPECTED:

```
{
  "Recommendations": {
    "4": ["66944", "21944", "812001", "87120", "312"],
    "14": ["66944", "21944", "812001", "87120", "312"],
    "16": ["66944", "21944", "812001", "87120", "312"],
    "65": ["66944", "21944", "812001", "87120", "312"],
    "66": ["66944", "21944", "812001", "87120", "312"],
    "67": ["66944", "21944", "812001", "87120", "312"]
  }
}
```

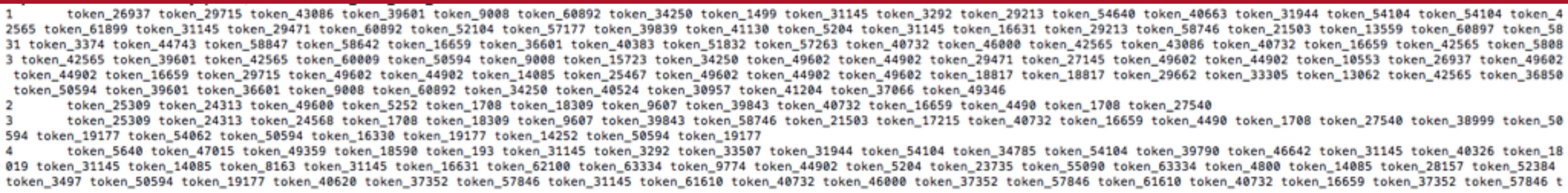
INPUT FILE 1



UID	SKU	Qty
4	46022	1.0
4	46334	1.0
14	66944	1.0
61	3905	3.0
61	39222	1.0
61	176395	1.0
65	169754	1.0
66	40859	7.0
66	41191	1.0
66	102067	1.0
70	38050	1.0
73	3771	1.0
73	38055	2.0

Gives information about the User transactions:
- UID (Unique Identifier for the user)
- SKU (Unique Identifier for the product)
- Qty (Quantity Purchased)

Approx Samples: 785M



```
1       token_26937 token_29715 token_43086 token_39601 token_9008 token_60892 token_34250 token_1499 token_31145 token_3292 token_29213 token_54640 token_40663 token_31944 token_54104 token_54104 token_4
2565 token_61899 token_31145 token_29471 token_60892 token_52104 token_57177 token_39839 token_41130 token_5204 token_31145 token_16631 token_29213 token_58746 token_21503 token_13559 token_60897 token_58
31 token_3374 token_44743 token_58847 token_58642 token_16659 token_36601 token_40383 token_51832 token_57263 token_40732 token_46000 token_42565 token_43086 token_40732 token_16659 token_42565 token_5808
3 token_42565 token_39601 token_42565 token_60009 token_50594 token_9008 token_15723 token_34250 token_49602 token_44902 token_29471 token_27145 token_49602 token_44902 token_10553 token_26937 token_49602
 token_44902 token_16659 token_29715 token_49602 token_44902 token_14085 token_25467 token_49602 token_44902 token_49602 token_18817 token_18817 token_29662 token_33305 token_13062 token_42565 token_36850
 token_50594 token_39601 token_36601 token_9008 token_60892 token_34250 token_40524 token_30957 token_41204 token_37066 token_49346
2       token_25309 token_24313 token_49600 token_5252 token_1708 token_18309 token_9607 token_39843 token_40732 token_16659 token_4490 token_1708 token_27540
3       token_25309 token_24313 token_24568 token_1708 token_18309 token_9607 token_39843 token_58746 token_21503 token_17215 token_40732 token_16659 token_4490 token_1708 token_27540 token_38999 token_50
594 token_19177 token_54062 token_50594 token_16330 token_19177 token_14252 token_50594 token_19177
4       token_5640 token_47015 token_49359 token_18590 token_193 token_31145 token_3292 token_33507 token_31944 token_54104 token_34785 token_54104 token_39790 token_46642 token_31145 token_40326 token_18
019 token_31145 token_14085 token_8163 token_31145 token_16631 token_62100 token_63334 token_9774 token_44902 token_5204 token_23735 token_55090 token_63334 token_4800 token_14085 token_28157 token_52384
token_3497 token_50594 token_19177 token_40620 token_37352 token_57846 token_31145 token_61610 token_40732 token_46000 token_37352 token_57846 token_61610 token_40732 token_16659 token_37352 token_57846 t
```

INPUT FILE 2

Gives information about the products purchased:

- SKU (Unique Identifier for the product)
- 'token_*****': Features for the product

The features are stemmed to provide uniformity

Approx Samples: 0.195M

PREPROCESSING

Due to the large size of the dataset, it will be advisable to pre-process it before starting any operations on it.

For Example:
all the features can be reduced from ‘token_id’ to only ‘id’ which would save you 43 bytes / token.
If we just calculate randomly: 43 bytes per token, 20 tokens per product and 0.195M token would be a large amount of space and even more harmful when it is going to be loaded in local memory.

PROCESSING

That is going to be your secret sauce.

Some libraries that may help you:
- https://github.com/josephmisiti/awesome-machine-learning (by josephmisiti)
A comprehensive list of all the online libraries available in each of the language. A must see.

- http://scikit-learn.org/stable/ (sci-kit learn)

- h20.ai (http://www.h2o.ai/)
Very powerful, faster and more accurate than sic-kit learn. Handles multi-core processing and data compression.

- Some more:
a) prediction.io
b) mldb.ai
c) Tensorflow
d) Apache Mahout
e) Algorithmia (Marketplace for Algorithms)

SUCCESS METRICS:

HIT RATE:
% of guests who bought at least one of the recommended items

PRECISION:
Number of items purchased from recommended items / Total number of recommendations

RECALL:
Number of items purchased from recommended items / Total number of items purchased

Points to Note:

- The recommendation should not include already bought products.
- Preferably, compute your accuracy using your training data itself using cross-validation. We do not provide a validation set.

PRIZES:

$10,000 for the Team that wins!

An OPTIONAL $5,000 for the second team if they come with something exceptional. *

QUESTIONS?

JAYKSHAH
@USC.EDU