WSM
FINAL PROJECT

第六組< wsm_dmmig>

110753201 曹昱維　110753132 馬行遠　110753207 林依樺　111753124 巫謹任

Outlines

01 Data Preprocessing

02 Training Model

03 Result

01

Data Preprocessing

a. Exploring Data

- In the item_features.csv file, there may be repeated featre_category_id in one item.

- feature_category 最大的重複次數(該feature_category 在同一個item中 會有複數值)
  - `less item_features.csv | awk -F "," '{print $1, $2}'|sort|uniq -d -c|awk -F " " '{if ($3== 1 ) {print $1, $3}}'|sort|uniq`
    - 將綠色的部份改成1 | 28 | 30 | 4 | 46 | 53
    - feature_category : max number of multiple value
      - 1 : 2
      - 28 : 3
      - 30 : 8
      - 4 : 4
      - 46 : 2
      - 53 : 2

b. Feature Engineering

1) <u>Filtering the datetime</u>

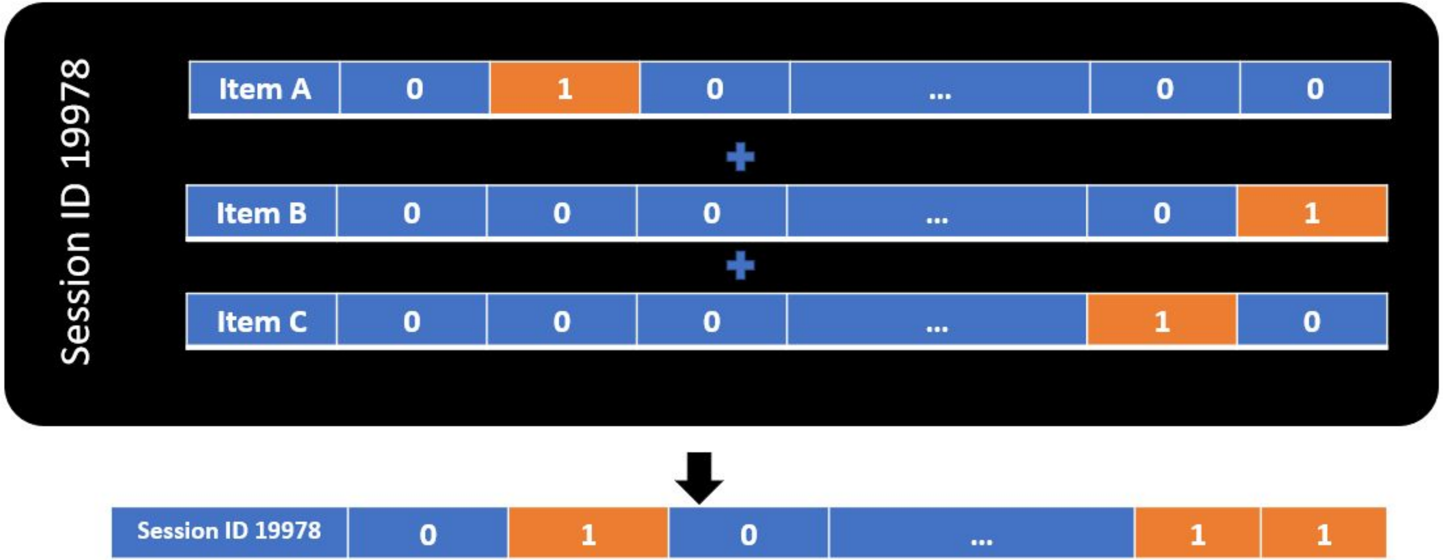
- Select the **session** of 2021/5/15~2021/5/31 to calculate TF-IDF, and the **candidate** part also selects the items that have appeared in 2021/5/15~2021/5/31

session_id	item_id	date	original_file
115	25976	2021-05-27 10:24:05.043	train_purchases
261	8840	2021-05-31 13:44:52.368	train_purchases
332	25415	2021-05-25 16:24:30.224	train_purchases
388	14800	2021-05-21 18:12:17.106	train_purchases
526	10915	2021-05-28 08:35:35.820	train_purchases
...	...	...	...
4439898	20891	2021-05-25 23:06:15.637	train_sessions
4439898	12508	2021-05-25 22:50:11.064	train_sessions
4439898	3237	2021-05-25 23:04:53.484	train_sessions
4439898	8414	2021-05-25 23:01:48.631	train_sessions
4439898	3237	2021-05-25 23:01:28.028	train_sessions

b. Feature Engineering

2) <u>Session Preprocess</u>: Linear superposition

- Combine all items in the same session to make a session into a vector.



b. Feature Engineering

3) <u>One-Hot-Encoding</u>:

   I. Feature expanded from **73 columns to 904 columns**

      e.g. if feature_category_id=1 and feature_category_value=60, the new feature name would be 1_60.

feature_category_id	feature_value_id	feature_name
1	60	1_60
1	143	1_143
1	358	1_358

b. Feature Engineering

3) <u>One-Hot-Encoding</u>:

   II. After processing multi-value, expand from **73 columns to 88 columns**

- If feature_category_id occurs twice in an item, the number of occurrences is listed after feature_category_id.
- e.g. if there are two feature_category_id=4 items in item 30, the number feature_category_id will be added to two columns: 4_1 and 4_2.

item_id	feature_category_id
30	16
30	57
30	4
30	68
30	61
30	8
30	55
30	4

→

item_id	feature_category_id
30	16_1
30	57_1
30	4_1
30	68_1
30	61_1
30	8_1
30	55_1
30	4_2

b. Feature Engineering

3) <u>One-Hot-Encoding</u>:

   III. Combine : **73+904=977**

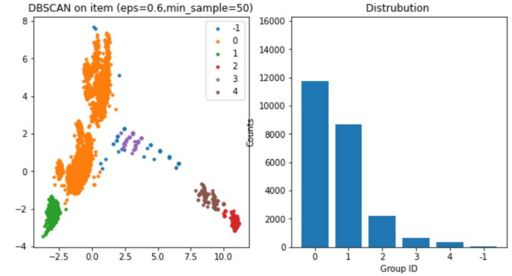
   - The results obtained in the preceding 1 are combined with the one-hot-encoding results of 73 features in the original data.

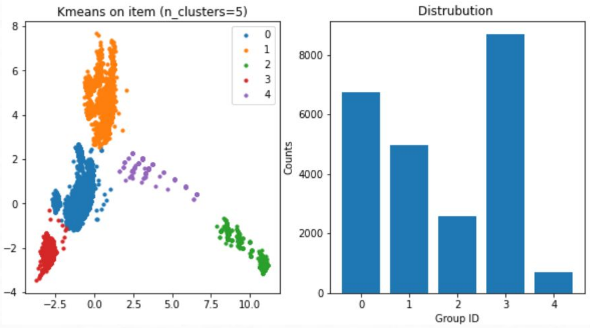
→ 23691 rows × 977 columns

	10_147	10_159	10_184	10_217	10_22	10_287	10_361	10_407	10_464	10_561	...	64	65	66	67	68	69	70	71	72	73
item_id																					
2	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	...	0.0	0.0	0.0	0.0	1.0	1.0	0.0	0.0	1.0	0.0
3	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	...	0.0	1.0	0.0	0.0	1.0	1.0	0.0	0.0	1.0	1.0

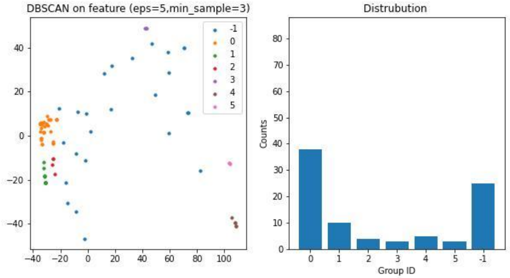
Clustering

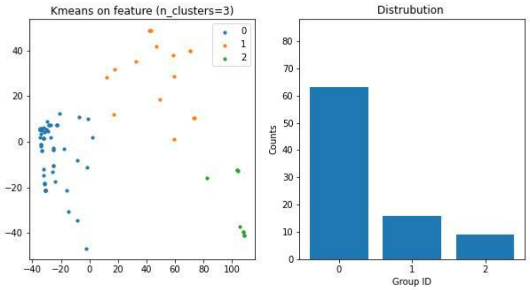
- **To find which items are more similar**





- **To find which features are more similar**





02

Training Model

Method1 TF-IDF

TF-IDF

- use **cosine similarity** to calculate the similarity.
- Each **Session** is treated as an **article (document)**

Session A	1	0	...	0	1
Session B	1	1	...	0	1

$$similarity(A,B) = \frac{A \cdot B}{\|A\| \times \|B\|}$$

Method 2 ITEM-CF

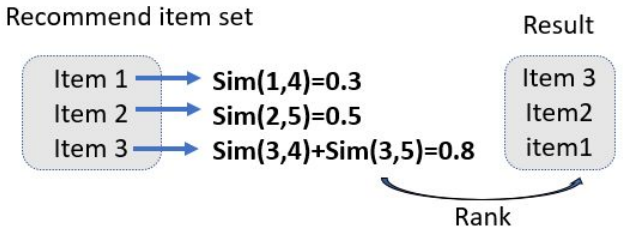
ITEM-CF

- each **session** as a **user**
- calculate the **similarity** of item *i* and item *j* when both of them co-occur in a user's list .
- Recommend by summarizing the similarity of all similar items for all viewed items by the user

item	1	2	3	4	5
1	1	0.1	0.1	0.3	0
2	0.1	1	0.2	0.3	0.5
3	0.1	0.2	1	0	0.5
4	0.3	0.3	0	1	0
5	0	0.5	0.5	0	1

Item similarity

User History	Similar items
Item 4 →	Item 1 ,Item 3
Item 5 →	Item 2 ,item 3



Method 3 Ensemble

Ensemble ITEM-CF & TF-IDF

- We combine itemcf and tfidf model into an ensemble model with a voting ratio R. The ratio indicates the contribution of two models.
- Re-rank recommended item with score:

$$\text{score}_i = 1 * \frac{1}{rank_{i,itemcf}} + R * \frac{1}{rank_{i,tfidf}}$$

$$\frac{1}{rank_{i,model}} = \begin{cases} \frac{1}{rank_{i,model}} & \text{if item } i \text{ exist} \\ 0 & \text{if item } i \text{ not exist} \end{cases}$$

03

Result

Results

ID	Method			Score
	Period of train data	Feature Engineer method	model	Leader Broad
1	2021/5/15~ 2021/5/31	• one-hot(904 columns) • session preprocess • Filtering the datetime	TF-IDF	0.04953
2	2021/5/1~ 2021/5/31	• one-hot(904 columns) • session preprocess • Filtering the datetime	TF-IDF	0.04867
3	2020/1/1~ 2021/5/31	• one-hot(904 columns) • session preprocess	TF-IDF	0.04770

Results

ID	Method			Score
	Period of train data	Feature Engineer method	model	Leader Broad
1	2020/1/1~2021/6/30 (only leaderboard)	• Filtering the by candidate	Item-CF with K=4	0.14584
2	2020/1/1~2021/6/30 (only leaderboard)	• Filtering the by candidate	Item-CF with K=2000	0.16909
3	2020/1/1~2021/6/30 (only leaderboard)		Item-CF with K=2000	0.16486
4	2021/1/1~2021/6/30 (only leaderboard)	• Filtering the by candidate • Filtering the datetime	Item-CF with K=2000	0.17068
5	2021/1/1~2021/6/30 (only leaderboard)	• Filtering the by candidate • Filtering the datetime	Item-CF with K=8000	0.17071

Results

ID	Method			Score	
	Period of train data	Feature Engineer method	model	Leader Broad	Final
1	2021/5/15~ 2021/5/31	• one-hot(904 columns) • session preprocess	TF-IDF	0.04953 980	
2	2021/1/1~ 2021/6/30 (only leaderboard)	• Filtering the by candidate • Filtering the datetime	Item-CF with K=8000	0.17071824222 743115	
3			Ensemble • ratio=0.02 • Item-CF with K=8000 • TF-IDF	0.1707183687116 8614	

Q & A

Thank you for your listening

04

Others

Method

Method

TF-IDF

- use **cosine similarity** to calculate the similarity.
- Each **Sesion** is treated as an **article (document)**

ITEM-CF

- each **session** as a **user**
- calculate the **similarity** of item *i* and item *j* when both of them co-occur in a user's list .