User data analysis: special issue of “Personal Data Market”

1. map from google play perm\_id into google default perm\_id
   1. save txt/permission\_table.xlsx $ perm\_jian table into txt file with name “perm\_jian\_input.txt”
   2. cm\_perm\_map.py: p\_map\_import(), perm\_map\_import()
2. get apps details:
   1. cm\_perm\_fact.py: perms\_fact\_perms\_map(), perms\_fact\_awards(), perms\_Fact\_apps(),
   2. cm\_contacts.py: contact\_videos(), contact\_share()
   3. cm\_util.py: categories\_get()
3. print out results:
   1. filter out apps: developer key is not null, perms key is not null, installs is less than 5, rating\_average is not 0, rating\_total is not 0
   2. save txt/permission\_table.xlsx$use\_data\_analysis table into txt file with name “user\_data\_analysis\_import\_txt”.
   3. output data filed:
      1. is\_theme\_app: 0=not theme, 1=theme\_keywords in app\_id, 2= theme\_keywords in app\_title, 3 = theme\_keywords in app\_desc
      2. award\_editor, award\_developer
      3. rating, install, price, developer
4. node graph diagram:
   1. map from permission to data type
5. Network graphic centrality:
   1. apps => permission
   2. permission => score
   3. calculate centrality score in bipartite mode: degree, betwenness, closeness.
   4. Check to see any data would be used more or less than 20% of average.

critical permission: ICIS 2013, Italy:

1. map from google play perm\_id into google default perm\_id
   1. save txt/permission\_table.xlsx $ perm\_jian table into txt file with name “perm\_jian\_input.txt”
   2. cm\_perm\_map.py: p\_map\_import(), perm\_map\_import()
2. get apps details:
   1. cm\_perm\_fact.py: perms\_fact\_perms\_map(), perms\_fact\_awards(), perms\_Fact\_apps(),
   2. cm\_contacts.py: contact\_videos(), contact\_share()
   3. cm\_util.py: categories\_get()
3. print out app results: app => perm\_id
   1. filter out apps: developer key is not null, perms key is not null, installs is less than 5, rating\_average is not 0, rating\_total is not 0
   2. out by category:
      1. for each category stored in : txt\_critical\_perm/apps\_file/%cate\_id\_%category.txt
      2. app\_id => perm\_id => category. The attribute should be complete.
4. generate GEXF file for visuallisation:
   1. Read each category app detail files from txt\_critical\_perm/apps\_file/%cate\_id\_%category.txt
   2. Add attributes, etc,
5. Calculate the centrality measure for each category:
   1. Generate timestap for each calculation.
   2. Read each category app detail files.
6. Proposition:
   1. Certain permissions are more valuable than other permission; as they are standing more lose to the central than others.
   2. The link between app and permission means mash up for innovation. Does not mean mashup too much would produce too much innovation/critical, does not mean mashup too little would produce too little. It is linked to the functionality behind.
   3. The link between app and permission means innovation or intrusitive. A app has more permission means more chance for innovation. However, at the same time, more permission also means more chance for intrusitiveness. So we will calculate the difference between certain permission here so see what patterns of permission would be more likely to contribute to innovation or intrusiveness based the normality central score.