process\_data():

*# 完成了两件事：  
# 1.处理原始文件，将内容信息存放在字典中并使用pickle封装；可以根据get\_mashup\_api\_id2info()接口获取信息；也可以通过get\_all\_texts()直接获得所有mashup和apis的text/tag信息  
# 2.将合法的mashup和api编码，生成调用矩阵。可以获取的信息有mashup\_num,api\_num,mashup\_api\_list等*

new\_split\_dataset.py

*# 将meta\_data.mashup\_api\_list划分成训练集和测试集  
# 然后得到对应的dataset对象*

dataset.py:  
1.meta\_data类：

内含process\_data对象，可以提供元信息的接口。

2. dataset类

*# 根据生成的训练测试集，生成一个dataset对象，里面实现了各种方法，更方便后续处理(train\_data,test\_data,his\_mashup\_ids的各种数据)  
# 访问时，使用dataset.crt\_ds 访问当前使用的对象  
# dataset.UV\_obj 访问根据当前训练集生成的get\_UV()对象，跟MF相关的数据*