一、收集

1.从github下载 `image-prediction.tsv` 2.从api读取json文件，按行存入 `tweet_json.txt` ，返回未读取的id和错误类型
3.‘Failed to send request’的错误类型不可复现，因此把未读取id再传入一次
4.‘'No status found with that ID.’类型原因是id无效，从expanded_urls列中提取与其对应的derived_id，与id结合为键值对传入amend_dict备用
5.将json中tweet_id,favorite_count,retweet_count传入dataframe查看
6.发现derived_id大都已经在dataframe中了，无需再次提取。是由'RT'造成的。

二、评估和清理

- `twitter-archive-enhanced` 中'doggo','floofer','pupper','puppo'应是‘stage’列的变量：
  在小写格式的text列中提取了关键字并拼接，将含有多个关键字的行标为'multiple',并验证了词提取覆盖了原先的数据无遗漏。
- `twitter-archive-enhanced` 中‘text’列包含了网址
  使用正则表达式提取text_url，并在原text中删除网址
- `twitter-archive-enhanced` 包含了一些转发的内容
  删除text以'RT @'开头的条目，打印出依旧带有‘RT’的text，发现是有效数据。删除空白
  的'retweeted_status_id','retweeted_status_user_id','retweeted_status_timestamp'三列
- `twitter-archive-enhanced` ‘timestamp’不是datetime datatype
  使用to_datetime转化
- `twitter-archive-enhanced` 'name'中有非名字的词
  经目测观察，无效名字的特征是小写字母开头，或者是字符形式'None'来表示无效值。找出这些无效值对应的text，发现可以提取named后面的词作为name列的值。对提取的值和欲替换的值进行比对，没问题后进行替换。并检查是否还有剩余的无效且非NaN的值。最后删掉辅助的named列。
- `twitter-archive-enhanced` 'name'等列中的缺失值被标记为字符形式的'None'
  此问题被上述清理一并解决
- `twitter-archive-enhanced` 'rating_denominator'有非10的项
  观察'rating_denominator'中非10的项，一种是没提取对位置，一种是对多个狗进行了评分。对所有数据重新提取：从text中截取10的倍数的分母及其分子，转化为float类型，放入rating_denominator和rating_numerator。为保留信息量，不对分子分母进一步处理，后续分析需注意排除异常值。
- `twitter-archive-enhanced` 'source'列有多余字符
  将'source'列用正则表达式分割为两列，source_name和source_href。
- `image-predictions` 'p1','p2','p3'大小写不统一
  使用capitalize()将p1 p1 p3全部转换成首字母大写
- 三个表格信息可合并
  使用merge()合并，以tweet_id作为主键，inner join