**疾病名称相似性度量方案**

1. **目的**

一个疾病名称有不同的说法，比如“肚子疼”，可以叫成“腹部痛”，“腹疼”，“肚子痛”等，关于疾病有很多种，有很多疾病名称，如果手动将每种疾病的名称分到相关疾病的目录下，那么工作量将十分巨大，所以需要一个自动将相关疾病名称分到相关疾病的下面。

**2．确定方案**

**2.1 前期准备工作**

疾病集合：symp\_info.hot 中第2列的34652个疾病。

主诉的集合：/data/nlp/corpus.predict\_disease/1.ori 中10185447条。

工具：symp\_extractor.py，将一条主诉中的疾病名称提取出来。

**2.2 方案步骤**

第一步，统计疾病与其他疾病在主诉集合中的频率。

|  |
| --- |
| **统计频率** |
| #用二维dict存储  #{disease\_1:{disease\_1\_1:f\_1\_1,…,disease\_1\_n:f\_1\_n},  disease\_2:{disease\_2\_1:f\_2\_1,…,disease\_2\_n:f\_2\_n},  , … ,  disease\_m:{disease\_m\_1:f\_m\_1,…,disease\_n\_n:f\_m\_n}}  for description in 主诉集合：  symp\_extractor.py提取疾病  如果有两个以上疾病，放入dict中，对应的频率加1. |

第二步，解析二维dict，相似的聚成一类。

1）反转二维的dict得到reverse\_dict，reverse\_dict的外围键为dict的内围键，内围键为dict的外围键，value为dict外围键的索引。

|  |
| --- |
| **反转二维dict** |
| #reverse\_dict需要将上述二维字典倒转，value值为最外层的索引  #{disease\_1\_1:{disease\_1:0,disease\_5:5,disease\_10:10,…,disease\_a:a},  disease\_1\_2:{ disease\_1:0,disease\_7:7,disease\_8:8,…,disease\_b:b },  ‘…’,  }  #已经实现 |

2）进行初步聚类，对于一个疾病名称，如果其字典中有*n*个疾病名称与另外一个疾病名称字典中疾病名称一样，暂时将他们归为一类。

|  |
| --- |
| **第一次聚类** |
| ‘’’  对reverse\_dicts进行组合，即从reverse\_dicts中随机取出n个键-值进行组合，对于每个组合，比如：  {key1:{key1\_1:val1\_1,key1\_2:val1\_2,…}}，  {key2:{key2\_1:val2\_1,key2\_2:val2\_2,…}}，  {key3:{key3\_1:val3\_1,key3\_2:val3\_2,…}}，  若内部词典中还有同名的疾病，比如key1\_1 = key2\_1 = key3\_1，key1\_2 = key2\_2 = key3\_2，则将key1\_1与key1\_2聚为一类。  ‘’’ |

3）对于pre\_dict中的每一个内部键，查找在dict中disease\_a\_b，disease\_c\_d，disease\_e\_f的键值，归一化生成特征向量：

如vector\_disease\_5 = [num\_a\_b,num\_c\_d,num\_e\_f]

4）在pre的内围字典中，根据上面得到的向量，两两计算疾病名称的向量距离dist，设定一个阈值dist\_threshold，如果：

dist < dist\_threshold

则认为他们是一类。