文本处理的第二步骤： 关键指标的建立

自然语言处理的统计定律：

Heaps’Law – estimating the number of terms: Token (单词) -> term(关键词)

The number of terms M in a collection

The number of tokens in the collection

$$M = kT^b$$

the number of tokens in the collection

$$\begin{aligned}\log M &= \log kT^b \\ &= \log T^b + \log k \\ &= b\log T + \log k\end{aligned}$$

constant

在文本集合中，Token 的数量越多，Term 的数量也就越多

Zipf’s law – modeling the distribution of terms

Then the collection frequency cf_i of the ith most common term is proportional to $1/i$.

$$cf_i \propto \frac{1}{i} \quad \text{or} \quad cf_i \cdot i = c$$

a constant

$$cf_i = ci^k$$

$k=-1$

a constant

and

$$\begin{aligned}\log cf_i &= \log ci^{-1} \\ &= -\log i + \log c\end{aligned}$$

排名越靠前的 Term,在文本中出现的频率越高

Term Frequency (TF)

- The weight of a term depends on the number of occurrences of the term in the document.
- Notation: $tf_{t,d}$ — the number of occurrences of term *t* in document *d*.

The bag of words model:

- The representation of a document *d* is the **set** of weights of its terms.

Document frequency:

- Notation: dft
- The number of documents in the collection that contain a term t.

Inverse Document Frequency

- For instance, a collection of documents on the auto industry is likely to have the term ‘*auto*’ in almost every document.

- We need a mechanism for reducing the effect of terms that **occur too often** in the collection.

Inverse document frequency (IDF):

Notation: $$idf_t = \log \frac{N}{df_t}$$ the number of documents in a collection

Log(N(文章总数)/ dft(term 在文章中的频率))

- The *idf* of a rare term is high, and is likely to be low for a frequent term.

Collection frequency (CF)

- The total number of occurrences of a term in the collection.

TF-IDF

assign the weight of term *t* in document *d*

- tf-idf$_{t,d}$ = tf$_{t,d}$ x idf$_t$.
- The weight of term t in document d is:
  - **High**, when t occurs many times in d and appears within a small number of documents.
  - **Low**, when t is a rare term in d and occurs in virtually all documents in the collection.

□ A simple scoring mechanism of a query q to a document d – the **overlap score measure**:

  - score(q,d) = $\sum_{t \text{ in } q}$tf-idf$_{t,d}$

$$\sum_{t \in dictionary} |(\underline{V}(d_1)_t - \underline{V}(d_2)_t)|$$

weight of term *t* in document d_2

□ Content-similar documents may have a significant vector difference <u>due to the **different document length**</u>.

Cosine similarity:

$$sim(d_1, d_2) = \frac{\underline{V}(d_1) \cdot \underline{V}(d_2)}{|\underline{V}(d_1)||\underline{V}(d_2)|}$$

inner product of vectors

$$\sum_{t \in dictionary} \underline{V}(d_1)_t * \underline{V}(d_2)_t$$

vector length

$$\sqrt{\sum_{t \in dictionary} \underline{V}(d_1)_t * \underline{V}(d_1)_t}$$

Variants in TF-IDF Functions：

□ **Sub-linear TF scaling**:

  - A common modification of TF is to use the <u>logarithm of the term frequency</u>.
  - Then, replace TF-IDF as WF-IDF:
    - □ *wf-idf$_{t,d}$* = *wf$_{t,d}$* * *idf$_t$*.

$$wf_{t,d} = \begin{cases} 1+\log tf_{t,d} & \text{if } tf_{t,d}>0 \\ 0 & \text{otherwise} \end{cases}$$

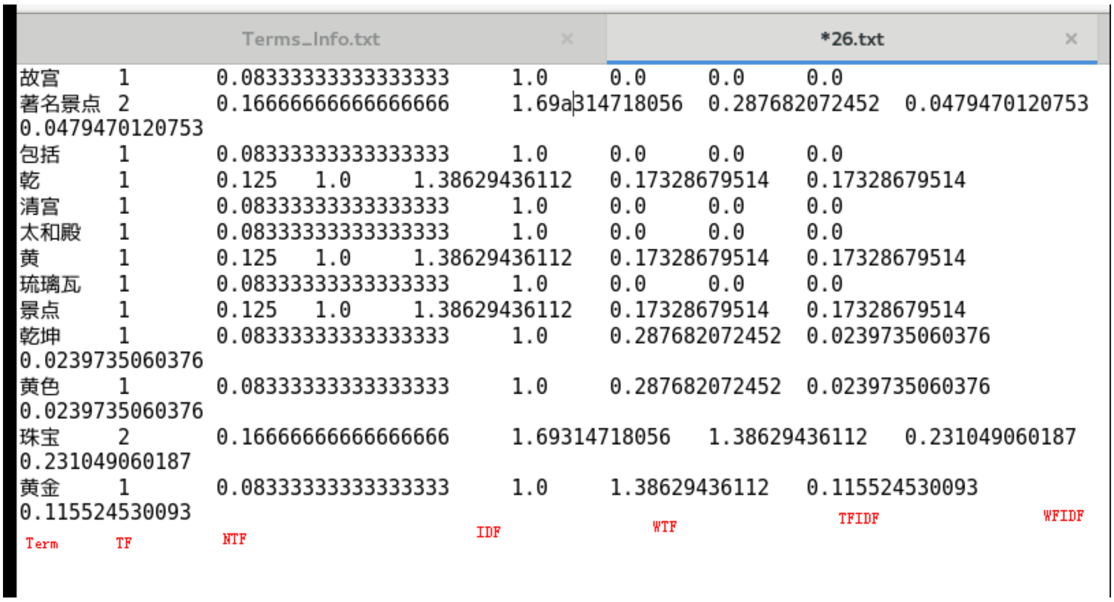
统计指标

- ND (Total Documents Number): 总文件数
- NTT (Total Terms (Type) Number)
- NTO (Total Terms (Occurrences) Number)
- TFt,d (The number of occurrences of term t in document d):文件 d 中 t 的出现次数,需要标准化
- DFt (The number of documents in the collection that contain a term t)
- CFt (The occurrences of term t in the collection)
- IDFt (Inverse document frequency)

  IDFt = log(N/DFt)
- TFtd-IDFtd= TFtd * IDFt
- WFtd-IDFtd= WFtd *IDFt

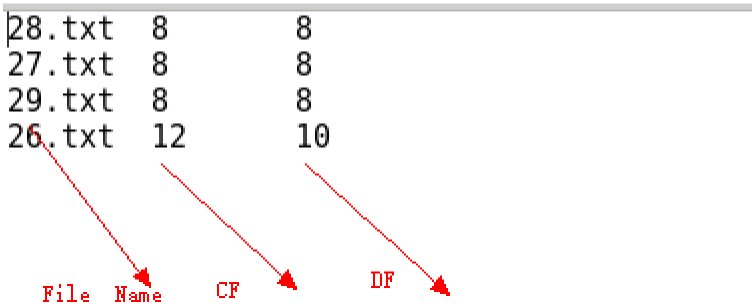
运行结果

单个文件的 term 统计结果



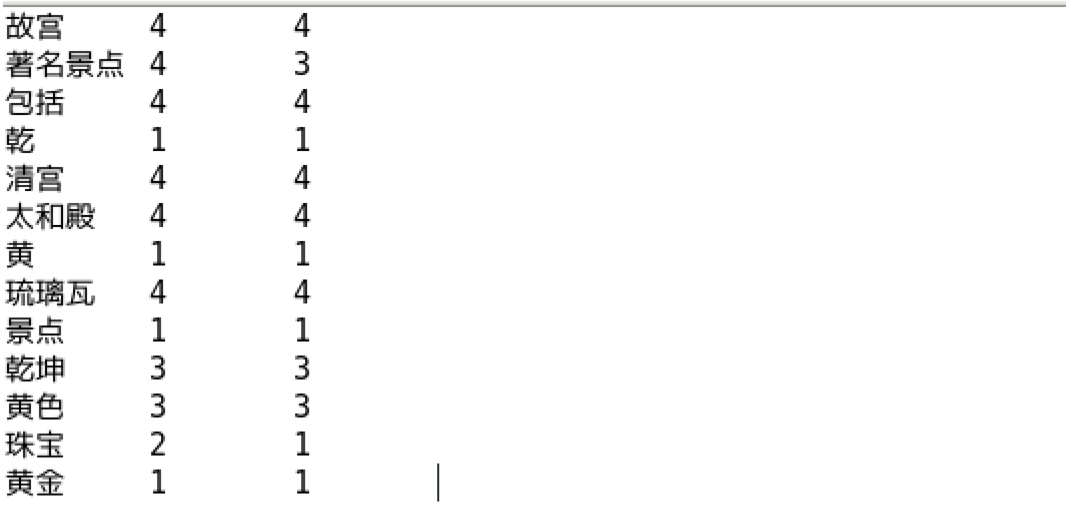
```
故宫      1       0.0833333333333333      1.0     0.0     0.0     0.0
著名景点  2       0.1666666666666666      1.69a314718056  0.287682072452  0.0479470120753
0.0479470120753
包括      1       0.0833333333333333      1.0     0.0     0.0     0.0
乾        1       0.125   1.0     1.38629436112   0.17328679514   0.17328679514
清宫      1       0.0833333333333333      1.0     0.0     0.0     0.0
太和殿    1       0.0833333333333333      1.0     0.0     0.0     0.0
黄        1       0.125   1.0     1.38629436112   0.17328679514   0.17328679514
琉璃瓦    1       0.0833333333333333      1.0     0.0     0.0     0.0
景点      1       0.125   1.0     1.38629436112   0.17328679514   0.17328679514
乾坤      1       0.0833333333333333      1.0     0.287682072452  0.0239735060376
0.0239735060376
黄色      1       0.0833333333333333      1.0     0.287682072452  0.0239735060376
0.0239735060376
珠宝      2       0.1666666666666666      1.69314718056   1.38629436112   0.231049060187
0.231049060187
黄金      1       0.0833333333333333      1.0     1.38629436112   0.115524530093
0.115524530093
```

各个文件的 Term 数量和种类统计



```
28.txt  8       8
27.txt  8       8
29.txt  8       8
26.txt  12      10
```

Term 的统计



```
故宫      4       4
著名景点  4       3
包括      4       4
乾        1       1
清宫      4       4
太和殿    4       4
黄        1       1
琉璃瓦    4       4
景点      1       1
乾坤      3       3
黄色      3       3
珠宝      2       1
黄金      1       1
```