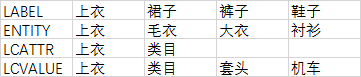
**Tag\_sentences.sh**

功能：根据每篇文章的题目，为文章打标签。

输入：

1. Tag\_tree：标签树，根据标签树建立标签字典，并且由此为文章打标签。



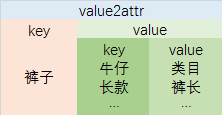
1. Sentences\_market：持久化后的文件，读入内存后是一个列表，每个元素是一个列表，包含两个元素，第1个是文章ID，第2个是文章标题。

输出：

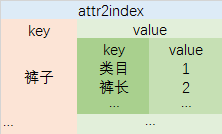
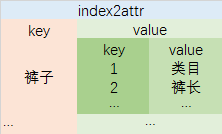
1. Tags\_market：标签文件，每一行是一篇文章，第1列是文章ID，第2列是类别标签，第3列是实体词，第4列是bitmap数组，数组每一个元素表示一个属性对应的值，例如值为5，二进制为101，表示包含这个属性的第1和第3个值。即两个标签<A,a1>,<A,a3>。



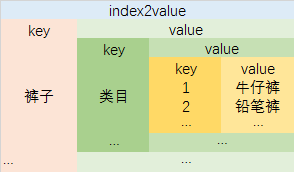
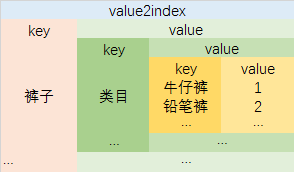
1. Dict\_market：字典文件，每一行是一个字典，第1列是字典名，第2列是字典。
   1. entity2label：实体对类别的映射，key是实体，value是类别。
   2. value2attr：属性和值得映射，key是类别，value是一个字典，这个字典的key是该类别下的所有属性的值得选项，value是值对应的属性。



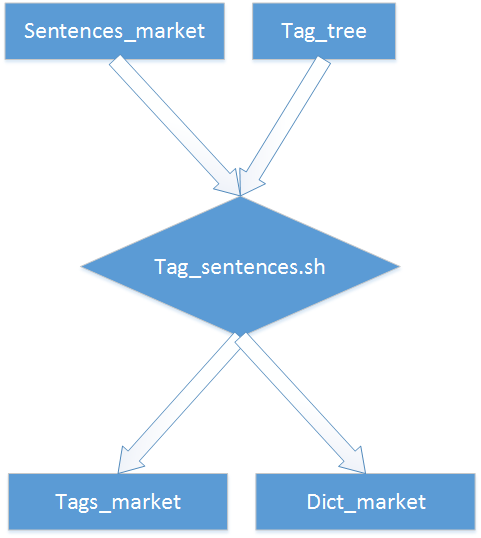
* 1. attr2index和index2attr：属性和索引的映射，key是类别，value是一个字典，这个字典的key是该类别下所有属性，value是对应的索引，后者则将key与value相反。

* 1. value2index和index2value：属性和索引的映射，key是类别，value是一个字典，这个字典的key是该类别下所有属性，value是一个字典，这个字典的key是该属性下所有的值选项，value是对应的索引，后者则将key与value相反。



流程图：

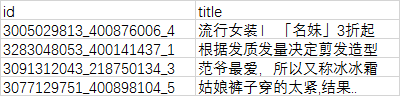


**Convert\_sentences.sh**

功能：将文章转化成pickle持久化，加速文件的读写。

输入：

1. Sentences：文章标题，每一行是一篇文章，第1列是文章ID，第2列是文章标题。



输出：

1. Sentences\_market：持久化后的文件，读入内存后是一个列表，每个元素是一个列表，包含两个元素，第1个是文章ID，第2个是文章标题。

流程图：

