大数据处理综合实验

实验2 倒排索引

实验报告

小组成员

尹浚宇 161130118

宋昱豪 171850505

刘扬 171850524

1.实验设计思路与伪代码

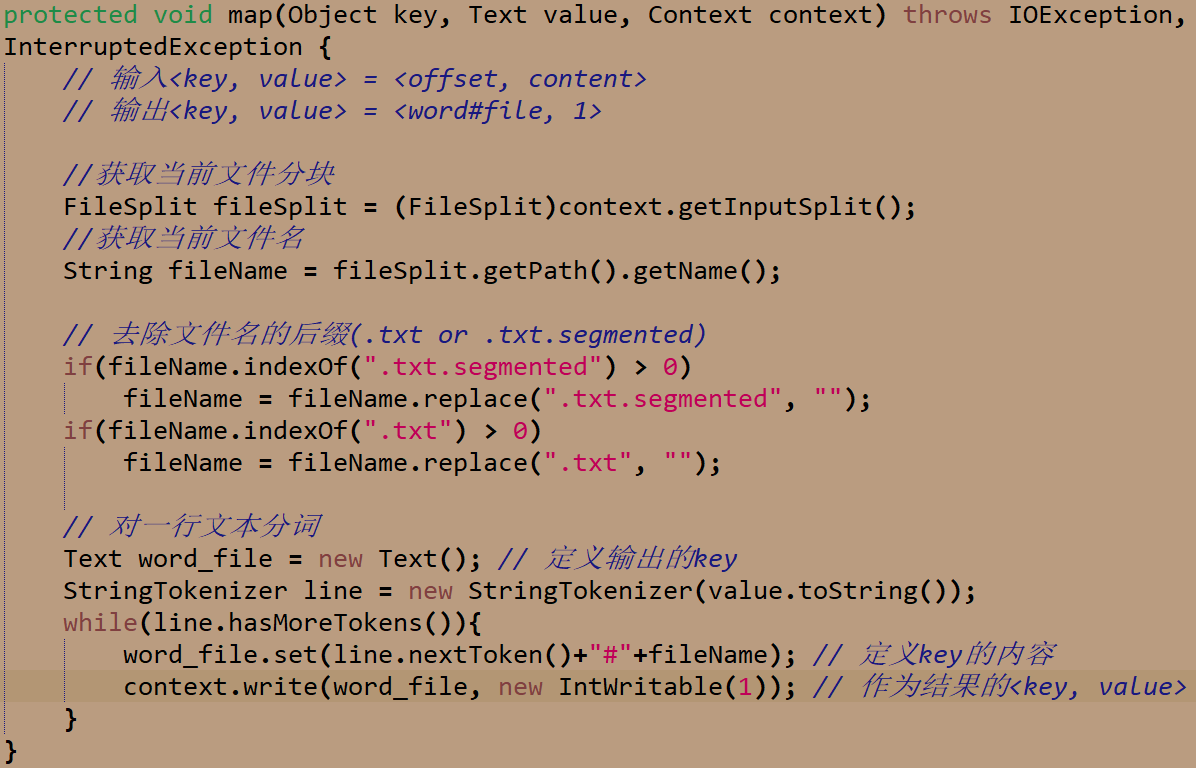
1.1基础任务

任务要求实现带词频属性的文档倒排索引算法, 并计算每个词语的平均出现次数; 两个计算任务在同一个Job中完成.

(1)Mapper设计

继承Mapper类重写map方法. Job指定输入格式为TextInputFormat, 所以map函数输入的key和value分别是当前行的偏移量和当前行的文本内容; 设计map输出的key和value分别是词语#文档名和词语计数.

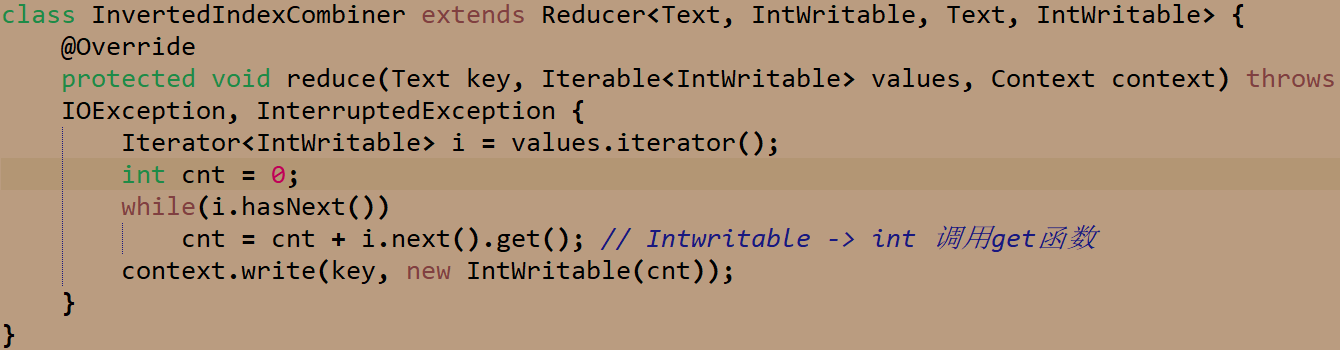
在map函数中, 先获取当前文档名, 去除后缀; 再对当前行进行分词; 最后根据定义好的输出格式拼接构成key和value.



(2)Combiner设计

为提高效率, 自定义Combiner对具有相同key的<key, value>进行合并.

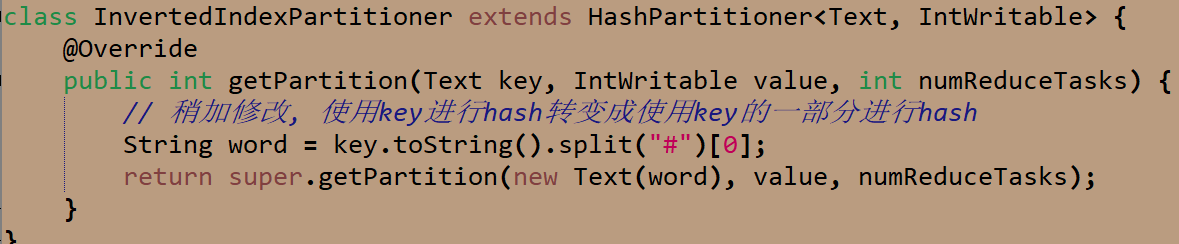
继承Reducer类并重写reduce方法. reduce方法的输入value是对应key的一组计数值, 目的是将这一组值进行求和构成一个value



(3)Partitioner设计

map方法输出的key是词语#文档名, 需要根据词语进行划分, 保证同一词语的<key, value>分配给同一个redeucer, 所以不能根据key进行划分, 需要自定义Partitioner根据词语进行划分.

继承HashPartitioner类重写getPartition方法. 从key中分理处词语, 将词语作为新的key, 调用基类的getPartition方法进行划分.



(4)Reducer设计

相比之下, Reducer的设计较为麻烦. 在上一步中, 已经保证同一个词语的<key, value>将发送至一个Reducer. 在Reducer中, 要对这些包含同一个词语的<key, value>进行统计.

继承Reducer类实现reduce和cleanup方法. reduce方法每次只能读取一个<key, value>, 但是由于key的定义, 同一个词语会对应多个<key, value>, 需要在类内定义数据成员作为缓冲区存储同一个词语的信息. 当读取到新的词语时, 代表前一个词语的信息已经全部统计完, 可以进行输出. 比较特殊的是最后一个词语, 最后一个词语的信息统计完成后, 由于没有下一个词语, 他的信息不会输出, 此时需要重写cleanup方法, 将最后一个词语的信息输出.



1.2 选做任务1

任务要求使用另外一个MapReduce Job对每个词语的平均出现次数进行全局排序，输出排序后的结果。设计的Job的输入基于基础实验的输出，即得到了所有词语及其出现的平均次数和其在各文件中出现的次数，对其对于词语在文件中出现的平均次数进行排序。

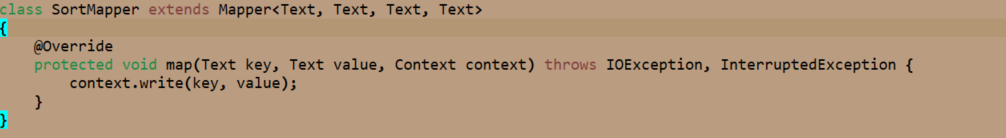
实验思路为使用已有的TotalOrderPartitioner类代替默认的HashPartitioner类, 对Mapper输出的key进行全局排序. 为了使用TotalOrderPartitioner类, 需要实现对数据进行取样, 对数据分布有个大致的了解. 使用InputSampler类进行实现取样, 取样结果存放在partitionFile.lst中供TotalOrderPartitioner类进行调用.

使用TotalOrderPartitioner类结合事先取样进行全局排序时, 取样只是获取key的分布, 且全局排序也是根据key进行排序的. 为了能够调用TotalOrderPartitioner类, 指定输入格式为KeyValueTextInputFormat，并且指定每一行key与value的分隔符是”,”, 这样便可以保证词频信息位于key中. 此时的key内容是词语和词频, 为了根据词频进行比较, 重写Comparator类, 对key进行处理, 分割得到词频并转换成浮点数表示, 然后进行比较. 最终实现全局排序的功能.

此任务对Mapper和Reducer类未进行较大改动, 只是为了配合TotalOrderPartitioner类的使用修改了输入格式和重写了用于比较的类.

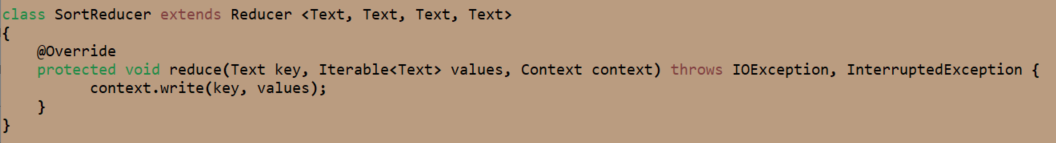
1. Mapper设计

由于基础实验已经计算完成，所以Map不需要对数据进行处理。



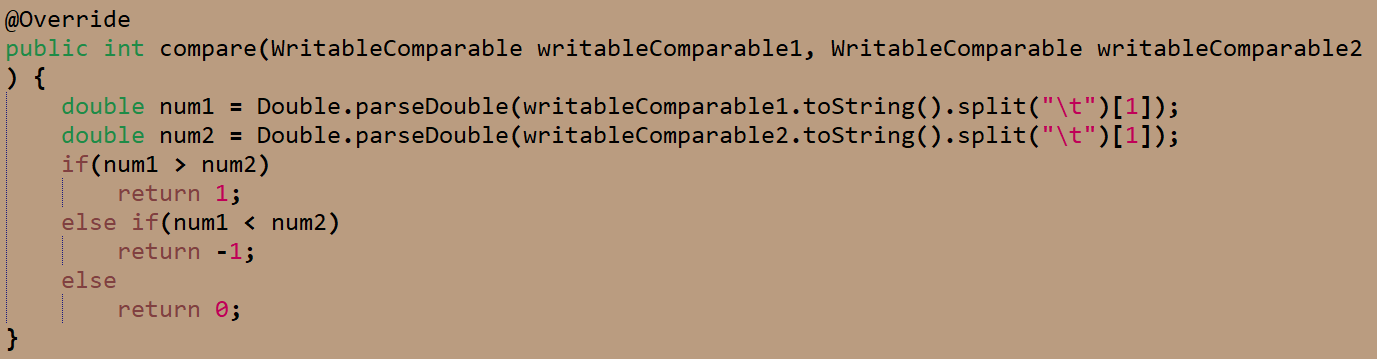
1. Reducer设计

由于基础实验已经计算完成，所以Reducer不需要对数据进行处理。



1. 重写Comparator

Comparator需要覆盖默认的比较函数，改为基于出现的平均次数进行比较。由于现在得到的键值对的格式就是基础实验的输出格式，所以对于输入的值，取split后的第二个单词，即为出现的平均次数，转化为数值类型后进行比较。



1.3 选做任务2

任务要求为每位作家计算每个词语的TF-IDF。TF定义为某个词语在某个作家的所有作品中的出现次数之和。IDF定义为。设置的任务的输入为基础实验的输入和输出。基础实验的输入用于计算文档的总数，从而便于统计IDF数据。

任务主要的操作是对基础任务输出内容进行处理. TF与IDF都可以根据基础任务输出内容的一行进行处理产生.

对于一个词语, 其IDF是固定的, 与作家无关. 所以对于一个词语, 每一个作家的IDF都是相同的. 可以事先计算, 不需要针对每个作家都求一次IDF.

TF的计算需要我们对于每一个文档名进行进一步分割. 基础任务输出的文档名是作者名和小说名拼接构成的, 此任务中需要我们进行进一步分割, 分别获取作者名和小说名, 从而便可得知对应一个作者的所有小说.

此任务中的一个小难点是获取文档总数. 做法是定义文件系统, 传入文档存储的文件夹, 便可以获取文档总数. 之后在conf变量中设置参数, 通过参数将文档总数传递给Mapper.

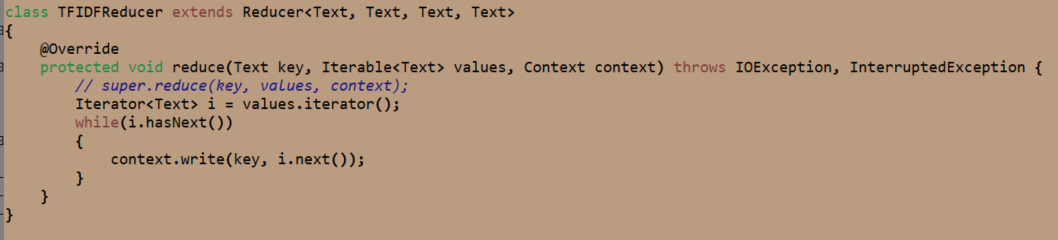
1. Mapper设计

基于基础实验的输出进行计算，使用了哈希表存储了作家和词语在其作品中出现的次数之和，即计算了TF。并基于文件数量计算了IDF，最后合并成输出写入。



1. Reducer设计

实际上不需要对数据进行过多处理，这里为了统一格式，对values重新进行了组织。



2.输出结果截图

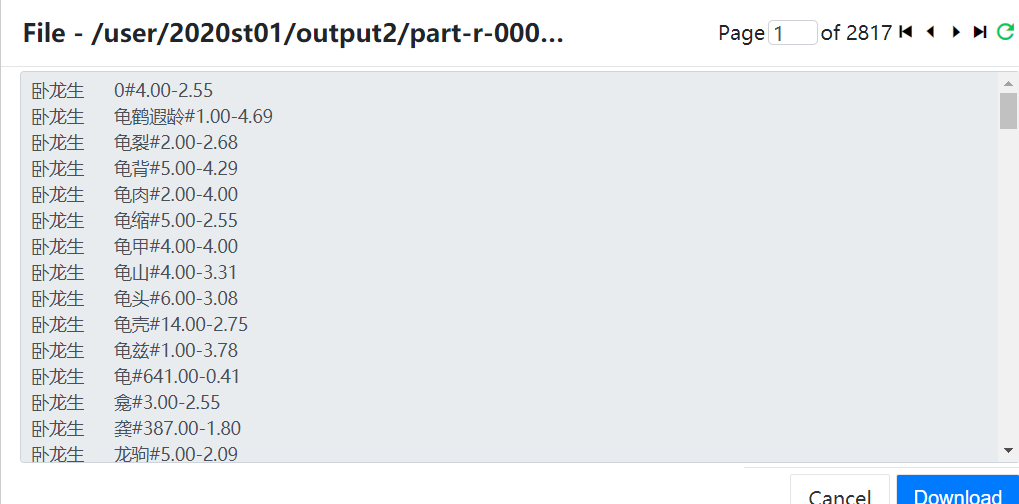
2.1 实验任务的输出文件截图



2.2 选做任务1的输出文件结果



2.3 选做任务2的输出文件结果



3.江湖 风雪的输出结果

3.1 江湖的输出结果

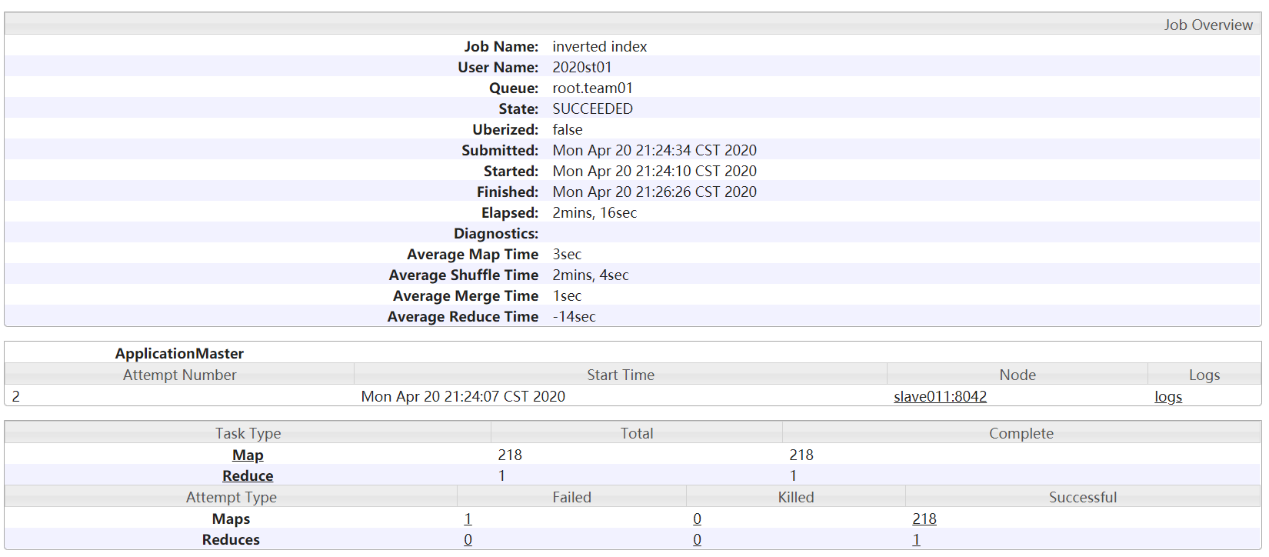
江湖 116.06, 卧龙生01.镖旗:275; 卧龙生02.春秋笔:329; 卧龙生03.翠袖玉环:402; 卧龙生04.地狱门:105; 卧龙生05.飞花逐月:298; 卧龙生06.飞铃:244; 卧龙生07.飞燕惊龙:269; 卧龙生08.风尘侠隐:228; 卧龙生09.风雨燕归来:198; 卧龙生10.黑白剑:223; 卧龙生11.黑白双娇:117; 卧龙生12.剑气洞彻九重天:299; 卧龙生13.剑无痕:261; 卧龙生14.剑仙列传:25; 卧龙生15.绛雪玄霜:274; 卧龙生16.金笔点龙记:318; 卧龙生17.金凤剪:326; 卧龙生18.金剑雕翎:397; 卧龙生19.惊鸿一剑震江湖:346; 卧龙生20.梦幻之刀:106; 卧龙生21.妙绝天香:28; 卧龙生22.女捕头:317; 卧龙生23.飘花令:263; 卧龙生24.七绝剑:130; 卧龙生25.七绝剑II还情剑:144; 卧龙生26.情剑无刃:7; 卧龙生27.琼楼十二曲:181; 卧龙生28.神州豪侠:261; 卧龙生29.双凤旗:261; 卧龙生30.素手劫:374; 卧龙生31.桃花血令:131; 卧龙生32.天鹤谱:106; 卧龙生33.天剑绝刀:321; 卧龙生34.天龙甲:293; 卧龙生35.天马霜衣:278; 卧龙生36.天香飙:246; 卧龙生37.天涯情侣:113; 卧龙生38.铁剑玉佩:71; 卧龙生39.铁苗神剑:358; 卧龙生40.无名箫:136; 卧龙生41.无形剑:256; 卧龙生42.新仙鹤神针:188; 卧龙生43.血剑丹心:350; 卧龙生44.烟锁江湖:238; 卧龙生45.燕子传奇:132; 卧龙生46.摇花放鹰传:446; 卧龙生47.一代天骄:353; 卧龙生48.银月飞霜:132; 卧龙生49.幽灵四艳:221; 卧龙生50.玉钗盟:320; 卧龙生51.玉手点将录:150; 卧龙生52.袁紫烟:30; 卧龙生53.岳小钗:308; 卧龙生54.指剑为媒:57; 古龙01.白玉老虎:111; 古龙02.白玉雕龙:24; 古龙03.碧血洗银枪:81; 古龙04.边城刀声:50; 古龙05.边城浪子:62; 古龙06.彩环曲:76; 古龙07.残金缺玉:82; 古龙08.苍穹神剑:131; 古龙09.楚留香01血海飘香:47; 古龙10.楚留香02大沙漠:23; 古龙11.楚留香03画眉鸟:96; 古龙12.楚留香04借尸还魂(鬼恋传奇):41; 古龙13.楚留香05蝙蝠传奇:80; 古龙14.楚留香06桃花传奇:8; 古龙15.楚留香07新月传奇:20; 古龙16.楚留香08午夜兰花:68; 古龙17.大地飞鹰:56; 古龙18.大旗英雄传:151; 古龙19.大人物:37; 古龙20.多情剑客无情剑:150; 古龙21.飞刀，又见飞刀:30; 古龙22.风铃中的刀声:76; 古龙23.孤星传:175; 古龙24.护花铃:193; 古龙25.欢乐英雄:64; 古龙26.浣花洗剑录:285; 古龙27.血鹦鹉:11; 古龙28.剑·花·烟雨·江南:18; 古龙29.剑毒梅香:110; 古龙30.剑客行:187; 古龙31.剑气书香:27; 古龙32.剑玄录:160; 古龙33.九月鹰飞:50; 古龙34.绝不低头:3; 古龙35.绝代双骄:249; 古龙36.猎鹰·赌局:107; 古龙37.流星·蝴蝶·剑:34; 古龙38.陆小凤01陆小凤前传（金鹏王朝）:15; 古龙39.陆小凤02绣花大盗:23; 古龙40.陆小凤03决战前后:26; 古龙41.陆小凤04银钩赌坊:10; 古龙42.陆小凤05幽灵山庄:40; 古龙43.陆小凤06凤舞九天:42; 古龙44.陆小凤07剑神一笑:47; 古龙45.名剑风流:201; 古龙46.那一剑的风情:32; 古龙47.怒剑狂花:102; 古龙48.飘香剑雨:67; 古龙49.七杀手:20; 古龙50.七星龙王:55; 古龙51.七种武器01长生剑:23; 古龙52.七种武器02碧玉刀:22; 古龙53.七种武器03孔雀翎:10; 古龙54.七种武器04多情环:8; 古龙55.七种武器05霸王枪:54; 古龙56.七种武器06离别钩:39; 古龙57.七种武器07拳头:16; 古龙58.情人箭:197; 古龙59.三少爷的剑:86; 古龙60.神君别传:12; 古龙61.失魂引:96; 古龙62.天涯·明月·刀:69; 古龙63.武林外史:179; 古龙64.湘妃剑:169; 古龙65.萧十一郎:56; 古龙66.火并萧十一郎:89; 古龙67.英雄无泪:54; 古龙68.游侠录:103; 古龙69.圆月弯刀:155; 古龙70.月异星邪:102; 李凉01.暗器高手:53; 李凉02.霸枪艳血:34; 李凉03.百败小赢家:79; 李凉04.本尊分身:97; 李凉05.超霸的男人:7; 李凉06.超级邪侠:209; 李凉07.赌棍小狂侠:31; 李凉08.公孙小刀:69; 李凉09.红顶记:48; 李凉10.滑头傻小子:43; 李凉11.会醉才会赢:189; 李凉12.活宝小淘气:121; 李凉13.江湖急救站:118; 李凉14.江湖双响炮:291; 李凉15.江湖一担皮:197; 李凉16.江湖一品郎:186; 李凉17.惊神关小刀:38; 李凉18.酒赌小浪子:17; 李凉19.酒狂任小赌:167; 李凉20.狂侠南宫鹰:15; 李凉21.六宝江湖行:72; 李凉22.矛盾天师:59; 李凉23.妙贼丁小勾:16; 李凉24.妙贼丁小勾续集:13; 李凉25.魔手邪怪:6; 李凉26.奇神扬小邪续集:98; 李凉27.奇神杨小邪:102; 李凉28.忍者龟:77; 李凉29.神偷小千:17; 李凉30.淘气世家:81; 李凉31.天才混混:89; 李凉32.天才混混外集:27; 李凉33.天齐大帝:62; 李凉34.天下第一当:52; 李凉35.武林嘻游记:180; 李凉36.小鬼大赢家:29; 李凉37.小鱼吃大鱼:42; 李凉38.笑笑江湖:69; 李凉39.新蜀山剑侠传:20; 李凉40.新蜀山剑侠传续:1; 李凉41.杨小邪发威:34; 梁羽生01.白发魔女传:63; 梁羽生02.冰川天女传:65; 梁羽生03.冰河洗剑录:65; 梁羽生04.冰魄寒光剑:4; 梁羽生05.草莽龙蛇传:143; 梁羽生06.大唐游侠传:97; 梁羽生07.弹指惊雷:77; 梁羽生08.飞凤潜龙:4; 梁羽生09.风雷震九洲:135; 梁羽生10.风云雷电:139; 梁羽生11.广陵剑:149; 梁羽生12.瀚海雄风:69; 梁羽生13.还剑奇情录:8; 梁羽生14.幻剑灵旗:27; 梁羽生15.慧剑心魔:102; 梁羽生16.剑网尘丝:127; 梁羽生17.江湖三女侠:252; 梁羽生18.绝塞传烽录:40; 梁羽生19.狂侠天娇魔女:180; 梁羽生20.联剑风云录:126; 梁羽生21.梁羽生传奇:74; 梁羽生22.龙凤宝钗缘:109; 梁羽生23.龙虎斗京华:182; 梁羽生24.鸣镝风云录:134; 梁羽生25.牧野流星:35; 梁羽生26.女帝奇英传:49; 梁羽生27.萍踪侠影录:40; 梁羽生28.七剑下天山:77; 梁羽生29.塞外奇侠传:2; 梁羽生30.散花女侠:48; 梁羽生31.随笔集：笔不花:4; 梁羽生32.随笔集：笔花六照:15; 梁羽生34.武当一剑:94; 梁羽生35.武林天骄:45; 梁羽生36.侠骨丹心:103; 梁羽生37.游剑江湖:275; 梁羽生38.云海玉弓缘:81; 金庸01飞狐外传:67; 金庸02雪山飞狐:29; 金庸03连城诀:35; 金庸04天龙八部:126; 金庸05射雕英雄传:57; 金庸06白马啸西风:5; 金庸07鹿鼎记:75; 金庸08笑傲江湖:199; 金庸09书剑恩仇录:64; 金庸10神雕侠侣:81; 金庸11侠客行:61; 金庸12倚天屠龙记:145; 金庸13碧血剑:65; 金庸14鸳鸯刀:35

3.2 风雪的输出结果

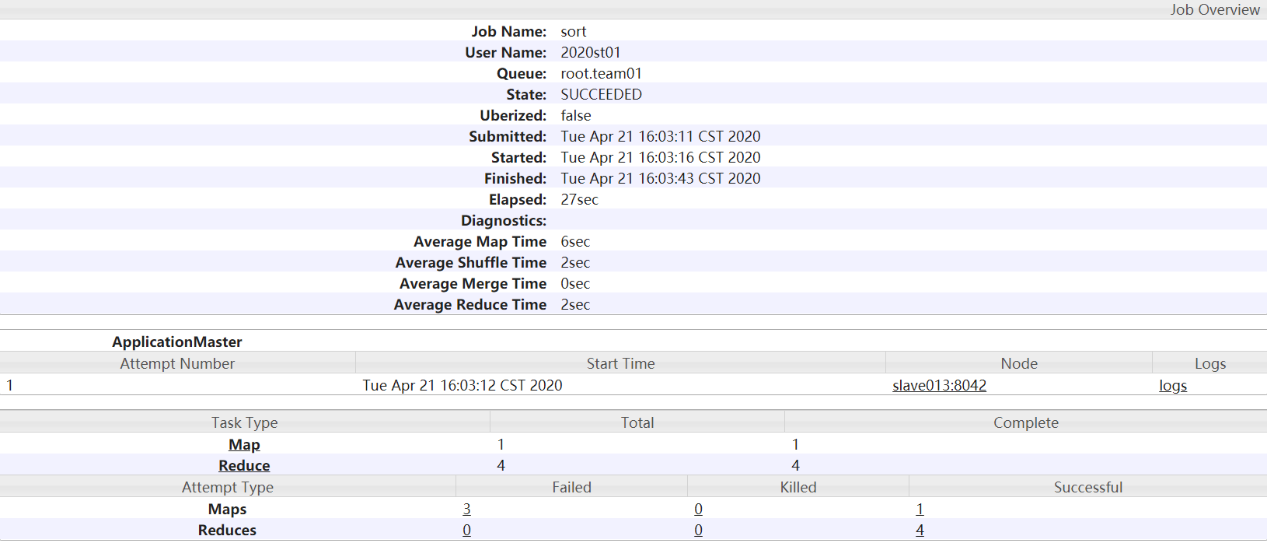
风雪 4.53, 卧龙生01.镖旗:3; 卧龙生07.飞燕惊龙:16; 卧龙生08.风尘侠隐:1; 卧龙生09.风雨燕归来:1; 卧龙生12.剑气洞彻九重天:36; 卧龙生15.绛雪玄霜:4; 卧龙生18.金剑雕翎:9; 卧龙生19.惊鸿一剑震江湖:2; 卧龙生22.女捕头:4; 卧龙生25.七绝剑II还情剑:2; 卧龙生27.琼楼十二曲:1; 卧龙生31.桃花血令:2; 卧龙生36.天香飙:2; 卧龙生38.铁剑玉佩:7; 卧龙生39.铁苗神剑:7; 卧龙生42.新仙鹤神针:15; 卧龙生47.一代天骄:3; 卧龙生49.幽灵四艳:1; 卧龙生52.袁紫烟:6; 古龙07.残金缺玉:7; 古龙13.楚留香05蝙蝠传奇:1; 古龙14.楚留香06桃花传奇:1; 古龙20.多情剑客无情剑:4; 古龙23.孤星传:2; 古龙24.护花铃:1; 古龙25.欢乐英雄:3; 古龙29.剑毒梅香:2; 古龙31.剑气书香:3; 古龙42.陆小凤05幽灵山庄:1; 古龙46.那一剑的风情:3; 古龙47.怒剑狂花:2; 古龙56.七种武器06离别钩:1; 古龙61.失魂引:14; 古龙63.武林外史:24; 古龙64.湘妃剑:5; 古龙67.英雄无泪:8; 李凉01.暗器高手:1; 李凉02.霸枪艳血:5; 李凉03.百败小赢家:1; 李凉04.本尊分身:6; 李凉09.红顶记:1; 李凉11.会醉才会赢:1; 李凉14.江湖双响炮:4; 李凉16.江湖一品郎:1; 李凉19.酒狂任小赌:32; 李凉31.天才混混:4; 李凉34.天下第一当:1; 李凉36.小鬼大赢家:1; 李凉39.新蜀山剑侠传:1; 李凉40.新蜀山剑侠传续:12; 梁羽生01.白发魔女传:4; 梁羽生02.冰川天女传:5; 梁羽生03.冰河洗剑录:1; 梁羽生07.弹指惊雷:1; 梁羽生09.风雷震九洲:1; 梁羽生10.风云雷电:3; 梁羽生12.瀚海雄风:1; 梁羽生14.幻剑灵旗:1; 梁羽生20.联剑风云录:1; 梁羽生21.梁羽生传奇:1; 梁羽生22.龙凤宝钗缘:5; 梁羽生25.牧野流星:5; 梁羽生26.女帝奇英传:3; 梁羽生27.萍踪侠影录:3; 梁羽生30.散花女侠:1; 梁羽生37.游剑江湖:1; 梁羽生38.云海玉弓缘:1; 金庸01飞狐外传:3; 金庸03连城诀:1; 金庸04天龙八部:1; 金庸05射雕英雄传:7; 金庸06白马啸西风:11; 金庸07鹿鼎记:1; 金庸10神雕侠侣:2; 金庸12倚天屠龙记:1

4. 执行报告

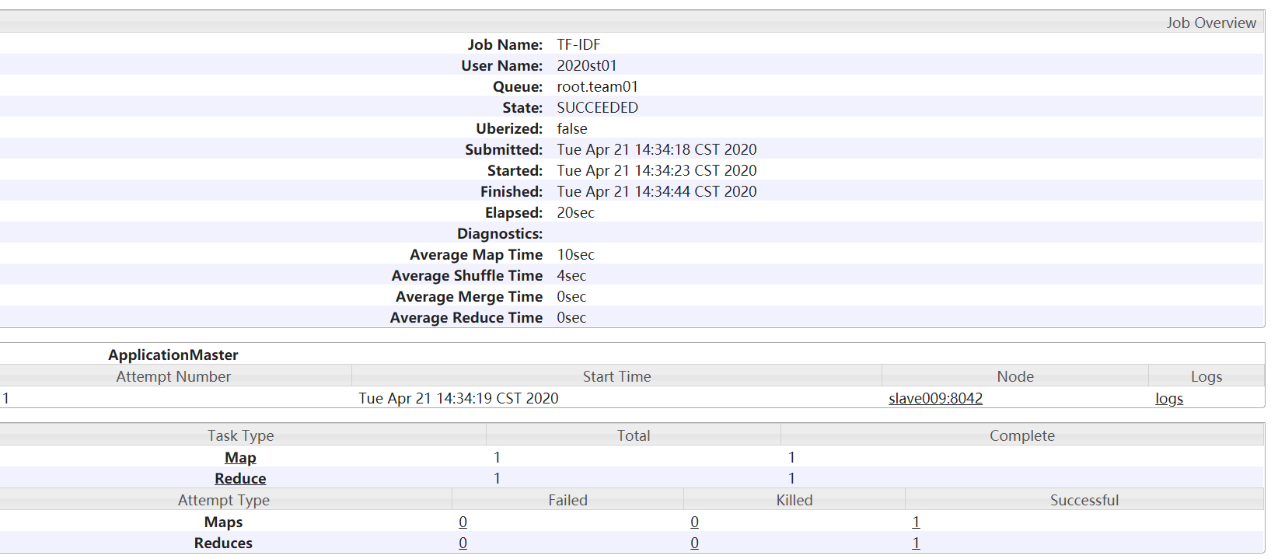
4.1 实验任务的WebUI执行报告



4.2 选做任务1的WebUI执行报告



4.3 选做任务2的WebUI执行报告



5.实验反思与不足

5.1 对于基础任务, 默认输入文件是已经进行的分词的文件. 事实上, 文件中包含一些我们不想要的词汇, 如C, com等与小说内容无关的词语. 我们并未对此进行过滤.

5.2 对于选做任务1, 任务要求实现全局排序. 使用一个Reducer和自定义Partitiioner都可以得到预期结果. 但是使用一个Reducer明显违背实验的目的, 自定义Partitioner类也有可能导致数据倾斜, 所以我们选择调用已有的TotalOrderPartitioner类进行全局排序. 为了配合这个类, 我们的输出文件格式与基本任务的输出文件格式略有不同, 具体而言是将词频后的逗号替换成了制表符. 我们曾在自定义Reducer类的reduce函数中尝试修改输出格式, 但是会触发key和value输出错位的bug.

6.实验分工

基础任务和两个选做任务的思路设计均是由小组三人共同讨论得出.

基础任务的代码实现: 刘扬

选做任务1的代码实现: 尹浚宇

选做任务2的代码实现: 宋昱豪

实验报告的撰写: 宋昱豪 刘扬

7.参考资料

<https://blog.csdn.net/qq_22271479/article/details/90740388?depth_1-utm_source=distribute.pc_relevant.none-task-blog-OPENSEARCH-3&utm_source=distribute.pc_relevant.none-task-blog-OPENSEARCH-3>

<https://blog.csdn.net/gywtzh0889/article/details/75392563>

<https://blog.csdn.net/qq_36864672/article/details/78586367?depth_1-utm_source=distribute.pc_relevant.none-task-blog-BlogCommendFromBaidu-1&utm_source=distribute.pc_relevant.none-task-blog-BlogCommendFromBaidu-1>