

选做功能 2：TF-IDF

一、设计思路

(1) Map 阶段

输入 Key-Value 类型：

- Key: `Object` (通常是文件的偏移量, 对于本程序无实际意义)
- Value: `Text` (代表文件的一行文本内容)

输出 Key-Value 类型：

- Key: `Text` (代表单词)
- Value: `Text` (格式为`<bookName>_<termFrequency>`, 其中`bookName`是文档名, `termFrequency`是该单词在文档中的频率)

思路：

1. 输入分割：`FileSplit`用来获取当前处理的文件片段。
2. 文件名获取：从`FileSplit`对象中获取当前文件的名称`bookName`。
3. 分词：使用`StringTokenizer`将每一行文本拆分成单词。
4. 计数：使用`HashMap`统计每个单词在当前文件中的频率。
5. 输出：对于每个单词，输出格式为`<单词, 文件名_单词频率>`。

(2) Reduce 阶段

输入 Key-Value 类型：

- Key: `Text` (代表单词)
- Value: `Text` (格式为`<bookName>_<termFrequency>`)

输出 Key-Value 类型：

- Key: `Text` (格式为`<bookName, 单词, 单词频率-IDF>`)
- Value: `Text` (空值, 仅用作占位符)

思路：

1. 初始化：从`Context`中获取总文档数`total`。
2. 处理：对于每个单词，汇总所有文档中的出现情况：

- 使用`HashMap`统计该单词在每个文档中的总频率。
 - 计算该单词在多少个文档中出现`docnum`。
3. IDF 计算: 使用公式 $IDF = \log_{10}(\text{总文档数} / (\text{包含该单词的文档数} + 1))$ 计算 IDF。
 4. 输出: 对于每个单词在每个文档中的 TF-IDF 值, 输出格式为`<bookName, 单词, 单词频率-IDF>`。

(3) 伪代码

MapReduce 中 Map 和 Reduce 的伪代码。

```
1 function map(Object key, Text value):
2     // 获取当前文件片段的文件名
3     fileSplit = getFileSplit(context)
4     bookName = getFileName(fileSplit)
5
6     // 初始化一个HashMap来记录单词频率
7     termFrequencyMap = new HashMap<String, Integer>()
8
9     // 将输入的文本行拆分为单词
10    tokenizer = new StringTokenizer(value.toString())
11    while tokenizer.hasMoreTokens():
12        word = tokenizer.nextToken()
13        if termFrequencyMap.containsKey(word):
14            termFrequencyMap.put(word, termFrequencyMap.get(word) + 1)
15        else:
16            termFrequencyMap.put(word, 1)
17
18    // 输出每个单词及其在当前文件中的频率
19    for each entry in termFrequencyMap:
20        term = entry.getKey()
21        frequency = entry.getValue()
22        context.write(new Text(term), new Text(bookName + "_" + frequency.toString()))
23
```

```
1 function setup(Context context):
2     // 从配置中获取总文档数
3     totalDocuments = Integer.parseInt(context.getConfiguration().get("total"))
4
5 function reduce(Text word, Iterable<Text> values, Context context):
6     term = word.toString()
7
8     // 初始化一个HashMap来记录每个文档中该单词的频率
9     tfMap = new HashMap<String, Integer>()
10
11    // 遍历values, 统计每个文档中该单词的频率
12    for each value in values:
13        parts = value.toString().split("_")
14        bookName = parts[0]
15        termFrequency = Integer.parseInt(parts[1])
16
17        if tfMap.containsKey(bookName):
18            tfMap.put(bookName, tfMap.get(bookName) + termFrequency)
19        else:
20            tfMap.put(bookName, termFrequency)
21
22    // 计算包含该单词的文档数
23    numDocsWithTerm = tfMap.size()
24
25    // 计算IDF
26    idf = log10((double) totalDocuments / (numDocsWithTerm + 1))
27
28    // 输出每个文档中该单词的TF-IDF值
29    for each entry in tfMap:
30        bookName = entry.getKey()
31        termFrequency = entry.getValue()
32        context.write(new Text(bookName + ", " + term + ", " + termFrequency + "-" + idf), new Text())
33
```

二、实验结果

输出文件路径：/resultTT/part-r-00000


(1) 输出结果文件的部分截图

```
shakespeare-coriolanus-24.txt, &C, 2-1.1249387366083
shakespeare-merry-15.txt, &C, 1-1.1249387366083
shakespeare-twelfth-20.txt, &c., 7-0.3233063903751334
shakespeare-tragedy-58.txt, &c., 1-0.3233063903751334
shakespeare-midsummer-16.txt, &c., 1-0.3233063903751334
shakespeare-as-12.txt, &c., 5-0.3233063903751334
shakespeare-pericles-21.txt, &c., 1-0.3233063903751334
shakespeare-loves-8.txt, &c., 1-0.3233063903751334
shakespeare-alls-11.txt, &c., 1-0.3233063903751334
shakespeare-merry-15.txt, &c., 3-0.3233063903751334
shakespeare-julius-26.txt, &c., 2-0.3233063903751334
shakespeare-much-3.txt, &c., 2-0.3233063903751334
shakespeare-hamlet-25.txt, &c., 1-0.3233063903751334
shakespeare-taming-2.txt, &c., 1-0.3233063903751334
shakespeare-tempest-4.txt, &c., 2-0.3233063903751334
shakespeare-third-53.txt, &c., 2-0.3233063903751334
shakespeare-coriolanus-24.txt, &c., 1-0.3233063903751334
shakespeare-titus-50.txt, &c., 1-0.3233063903751334
shakespeare-second-52.txt, &c., 1-0.3233063903751334
shakespeare-life-55.txt, &c., 3-0.3233063903751334
shakespeare-hamlet-25.txt, &c.', 1-1.1249387366083
shakespeare-julius-26.txt, &c.', 1-1.1249387366083
shakespeare-alls-11.txt, &c., 1-1.0
shakespeare-titus-50.txt, &c., 2-1.0
shakespeare-life-55.txt, &c., 1-1.0
shakespeare-antony-23.txt, &c], 1-0.4881166390211256
shakespeare-hamlet-25.txt, &c], 3-0.4881166390211256
shakespeare-timon-49.txt, &c], 2-0.4881166390211256
shakespeare-tragedy-57.txt, &c], 1-0.4881166390211256
shakespeare-tempest-4.txt, &c], 1-0.4881166390211256
shakespeare-othello-47.txt, &c], 1-0.4881166390211256
shakespeare-romeo-48.txt, &c], 2-0.4881166390211256
shakespeare-coriolanus-24.txt, &c], 1-0.4881166390211256
```

```
shakespeare-macbeth-46.txt, further., 3-0.37161106994968846
shakespeare-merry-15.txt, further., 1-0.37161106994968846
shakespeare-julius-26.txt, further., 2-0.37161106994968846
shakespeare-hamlet-25.txt, further., 1-0.37161106994968846
shakespeare-cymbeline-17.txt, further., 1-0.37161106994968846
shakespeare-tempest-4.txt, further., 1-0.37161106994968846
shakespeare-othello-47.txt, further., 1-0.37161106994968846
shakespeare-coriolanus-24.txt, further., 7-0.37161106994968846
shakespeare-life-55.txt, further., 2-0.37161106994968846
shakespeare-antony-23.txt, further., 1-0.5606673061697374
shakespeare-cymbeline-17.txt, further., 1-0.5606673061697374
shakespeare-timon-49.txt, further., 1-0.5606673061697374
shakespeare-merchant-5.txt, further., 1-0.5606673061697374
shakespeare-midsummer-16.txt, further., 1-0.5606673061697374
shakespeare-coriolanus-24.txt, further., 1-0.5606673061697374
shakespeare-pericles-21.txt, further., 1-0.5606673061697374
shakespeare-life-54.txt, further., 1-0.5606673061697374
shakespeare-macbeth-46.txt, further., 2-0.5606673061697374
shakespeare-merry-15.txt, further., 1-0.5606673061697374
shakespeare-first-51.txt, further., 1-0.6478174818886375
shakespeare-cymbeline-17.txt, further., 1-0.6478174818886375
shakespeare-timon-49.txt, further., 1-0.6478174818886375
shakespeare-third-53.txt, further., 1-0.6478174818886375
shakespeare-othello-47.txt, further., 1-0.6478174818886375
shakespeare-alls-11.txt, further., 1-0.6478174818886375
shakespeare-macbeth-46.txt, further., 1-0.6478174818886375
shakespeare-julius-26.txt, further., 1-0.6478174818886375
shakespeare-hamlet-25.txt, further?, 1-0.9030899869919435
shakespeare-timon-49.txt, further?, 1-0.9030899869919435
shakespeare-alls-11.txt, further?, 1-0.9030899869919435
shakespeare-life-55.txt, further?, 1-0.9030899869919435
shakespeare-pericles-21.txt, furtherance, 1-1.1249387366083
shakespeare-life-54.txt, furtherance, 1-1.1249387366083
```

shakespeare-life-54.txt, painted, 3-0.1106982974936897
shakespeare-macbeth-46.txt, painted, 1-0.1106982974936897
shakespeare-merry-15.txt, painted, 1-0.1106982974936897
shakespeare-winters-19.txt, painted, 1-0.1106982974936897
shakespeare-taming-2.txt, painted, 3-0.1106982974936897
shakespeare-coriolanus-24.txt, painted, 1-0.1106982974936897
shakespeare-second-52.txt, painted, 1-0.1106982974936897
shakespeare-two-18.txt, painted, 1-0.1106982974936897
shakespeare-first-51.txt, painted, 1-0.1106982974936897
shakespeare-venus-60.txt, painted, 1-0.1106982974936897
shakespeare-troilus-22.txt, painted, 2-0.1106982974936897
shakespeare-sonnets-59.txt, painted, 5-0.1106982974936897
shakespeare-rape-61.txt, painted, 4-0.1106982974936897
shakespeare-midsummer-16.txt, painted, 2-0.1106982974936897
shakespeare-life-56.txt, painted, 2-0.1106982974936897
shakespeare-as-12.txt, painted, 2-0.1106982974936897
shakespeare-loves-8.txt, painted, 3-0.1106982974936897
shakespeare-julius-26.txt, painted, 1-0.1106982974936897
shakespeare-hamlet-25.txt, painted, 2-0.1106982974936897
shakespeare-cymbeline-17.txt, painted, 1-0.1106982974936897
shakespeare-tempest-4.txt, painted, 1-0.1106982974936897
shakespeare-third-53.txt, painted, 1-0.1106982974936897
shakespeare-sonnets.txt, painted, 4-0.1106982974936897
shakespeare-life-55.txt, painted, 1-0.1106982974936897
shakespeare-titus-50.txt, painted, 2-0.1106982974936897
shakespeare-two-18.txt, painted,, 1-0.8239087409443188

(2) Yarn Resource Manager 的 WebUI 执行报告内容



Application application_1678703754602_10077

Logged in as: drs@ha

Cluster

About
Nodes
Node Labels
Applications

NEW
NEW SAVING
SUBMITTED
ACCEPTED
RUNNING
FINISHED
KILLED
Scheduler

Tools

Application Overview

User: 2024tsu_10

Name: TRIOF

Application Type: MAPREDUCE

Application Tags:

Application Priority: 0 (Higher Integer value indicates higher priority)

YarnApplicationState: FINISHED

Queue: 2024Mapdata@tsu2

FinalStatus Reported by AM: SUCCEEDED

Started: Tue May 14 10:23:33 +0800 2024

Launched: Tue May 14 10:27:02 +0800 2024

Finished: Tue May 14 10:27:44 +0800 2024

Elapsed: 4mins, 11sec

Tracking URL: History

Log Aggregation Status: RUNNING

Application Timeout (Remaining Time): Unlimited

Diagnostics:

Unmanaged Application: false

Application Node Label expression: <Not set>

AM container Node Label expression: <DEFAULT_PARTITION>

Application Metrics

Total Resource Preempted: <memory@, vCores@>

Total Number of Non-AM Containers Preempted: 0

Total Number of AM Containers Preempted: 0

Resource Preempted from Current Attempt: <memory@, vCores@>

Number of Non-AM Containers Preempted from Current Attempt: 0

Aggregate Resource Allocation: 824483 MB-seconds, 535 vcore-seconds, 0 yarn.io/gpu-seconds

Aggregate Preempted Resource Allocation: 0 MB-seconds, 0 vcore-seconds

Show 20 entries

Attempt ID	Started	Node	Logs	Nodes blacklisted by the app	Nodes blacklisted by the system
attempt_1678703754602_10077_000002	Tue May 14 10:27:02 +0800 2024	https://view009.8042	Logs	0	0
attempt_1678703754602_10077_000001	Tue May 14 10:23:42 +0800 2024	https://view004.8042	Logs	0	0

Showing 1 to 2 of 2 entries

First Previous 1 Next Last