

数据库原理（1）期末报告

小组成员	
姓名	学号
汪江豪	22121630
董晓鸥	22121397

一、 题目选择：

实验 4: Spark-SQL 分布式数据库查询-推荐系统（大数据集）

二、 实验思路：

使用两个中间表 `ref_count` 记录每个网站的外部标签数；`common` 记录任意两个网站的公共标签数量。并集 $|A \cup B| = |A| + |B| - |A \cap B|$ ，最后交集 \div 并集即为两个网站之间的相似度。

三、 实验步骤：

1. 准备工作：

与实验 4 步骤基本相同，在腾讯云上租用 3 台云服务器，先搭建好基于 hadoop 的分布式文件系统，在 hadoop 目录下运行 `sbin/start-all.sh`，启动 hadoop，三台节点 `master`, `slave01`, `slave02`，查询 `jps`，确保 `datanode` 等信息出现。

2. 设置工作节点：

(1) 将 `slave01`, `slave02` 设为工作节点，`master` 默认作为任务调度和资源分配节点。以下给出方法：在 `spark` 目录下，`cd conf`，打开 `spark-env.sh`，在头部新增 `export HADOOP_CONF_DIR=/usr/local/hadoop/etc/hadoop`

将 hadoop 下的配置文件保存到 spark 配置文件的环境变量中，以便启动 `spark-sql` 时使用资源分配命令。

3. 启动 spark-sql，将 slave01, slave02 设为工作节点：

在 `spark` 目录下运行命令：

```
bin/spark-sql -master yarn -num-executors 2
```

该命令指明启动 `spark-sql` 时，选择 `slave01`, `slave02` 作为工作节点。

4. 将大数据集上床到 hdfs 上后，使用如下命令导入大数据集，并创建为 relation 表，列属性名为 `referrer`, `referree`，数据类型为 `int`。

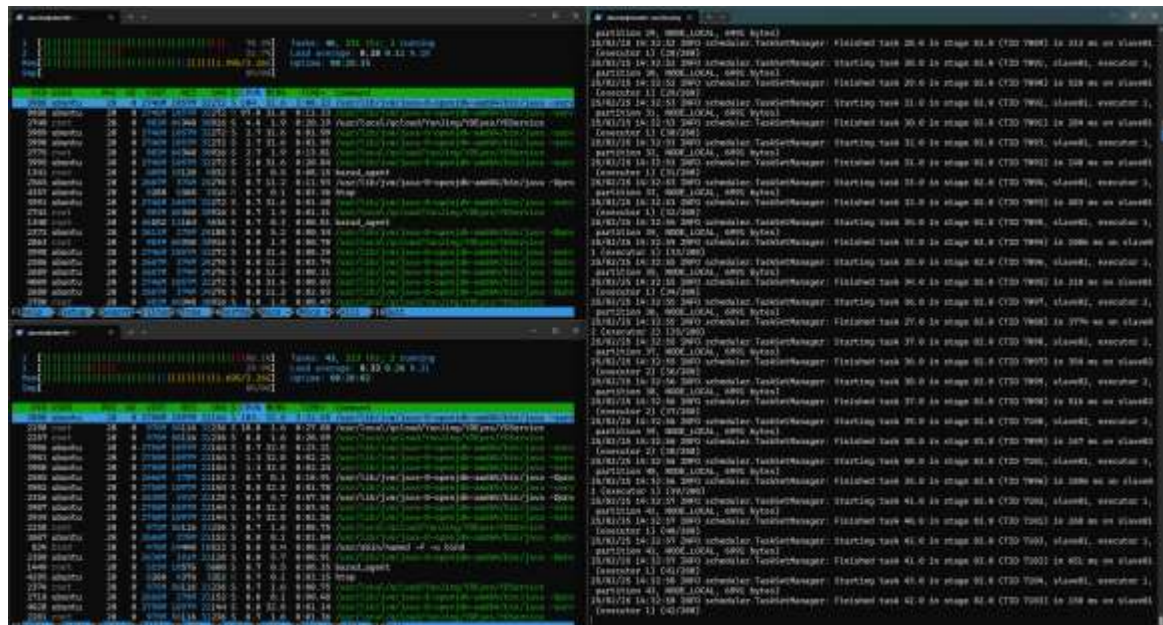
```
create table relation(  
  referrer int,  
  referree int  
)  
using csv  
options(  
  path './large_relation',  
  delimiter ' ',  
  header 'false'  
);
```

5. 使用连接方式进行查询：

语句如下，由于结果有 6400 万多行，为了节省数据打印时间，这里限制返回最多 500 行：

```
with ref_count as (select referrer, count(distinct referree) as
web_count
    from relation group by referrer),
common as (select a.referrer as web1, b.referrer as web2, count(*) as
com_cnt
    from relation a join relation b
    on a.referree = b.referree
    where a.referrer < b.referrer
    group by a.referrer, b.referrer)
select common.web1 as web1, common.web2 as web2,
case when r1.web_count + r2.web_count - common.com_cnt > 0
then common.com_cnt / (r1.web_count + r2.web_count - common.com_cnt)
else 0
end as similarity
from common join ref_count r1 on common.web1 = r1.referrer
join ref_count r2 on common.web2 = r2.referrer
order by similarity desc limit 500;
```

运行状况截图：



截图显示：在执行查询过程中，slave01,slave02 共 4 个 CPU 核心均处于工作状态，表明查询计算成功分布在两台机器上。

结果展示：

```

19298936      437136030      1.0
2603697 22817535      1.0
15108825      80863675      1.0
27377864      158592175      1.0
14467975      97742874      1.0
16884125      128953441      1.0
42437419      89894179      1.0
98889462      159570745      1.0
25988804      437136030      1.0
27889544      86525478      1.0
24889018      105639498      1.0
88228895      437136030      1.0
236425217      536055930      1.0
22558264      105639498      1.0
224849849      285949147      1.0
14467975      285949147      1.0
14051888      52659511      1.0
42988746      89894179      1.0
326386597      437136030      1.0
17455041      345999006      1.0
14768311      52659511      1.0
15387913      86376769      1.0
337414203      345999006      1.0
12742918      285949147      1.0
26479461      437136030      1.0
39381817      77016942      1.0
14637838      22299596      1.0
Time taken: 114.652 seconds, Fetched 500 row(s)
25/02/25 14:34:39 INFO CliDriver: Time taken: 114.652 seconds, Fetched 500 row(s)
spark-sql> |

```

耗时 114.652 秒完成。

四、 更多方式尝试

为了比较不同方式速度差异，我将实验 4 的其余几种查询语句也进行了测试，sql 语句以及结果如下：

- 嵌套相关子查询

查询语句：

```

WITH ref_count AS (
    SELECT referrer, COUNT(DISTINCT referree) AS web_count
    FROM relation
    GROUP BY referrer
),
common AS (
    SELECT
        a.referrer AS web1,
        b.referrer AS web2,
        COUNT(*) AS com_cnt,
        min((SELECT web_count FROM ref_count WHERE referrer =
a.referrer)) AS web1_count,
        min((SELECT web_count FROM ref_count WHERE referrer =
b.referrer)) AS web2_count
    FROM relation a
    JOIN relation b ON a.referree = b.referree
    WHERE a.referrer < b.referrer
    GROUP BY a.referrer, b.referrer
)
SELECT
    common.web1,
    common.web2,
    CASE

```

```

        WHEN common.web1_count + common.web2_count - common.com_cnt >
0
        THEN common.com_cnt / (common.web1_count + common.web2_count -
common.com_cnt)
        ELSE 0
    END AS similarity
FROM common ORDER BY similarity DESC LIMIT 500;

```

查询结果:

```

32423090      37650796      1.0
15315990      15866010      1.0
15070088      37650796      1.0
15821614      15866010      1.0
16381285      37650796      1.0
15144128      15941344      1.0
22362074      37650796      1.0
15204398      15941344      1.0
15315990      37650796      1.0
6343638 15941344      1.0
16948499      37650796      1.0
15490539      16012228      1.0
15841269      37650796      1.0
14536584      16012228      1.0
24719498      37650796      1.0
15490539      16009782      1.0
15445317      37650796      1.0
15679117      16154129      1.0
21964700      37650796      1.0
15487600      16296160      1.0
19542313      37650796      1.0
5706198 16479292      1.0
15821614      37650796      1.0
13485638      17017895      1.0
17198379      37650796      1.0
60769 17017895      1.0
15729027      37871899      1.0
Time taken: 156.646 seconds, Fetched 500 row(s)
25/02/25 14:39:47 INFO CliDriver: Time taken: 156.646 seconds, Fetched 500 row(s)
spark-sql> |

```

耗时 156.646s 完成。

- 嵌套不相关子查询:

查询语句:

```

WITH ref_count AS (
    SELECT referrer, COUNT(DISTINCT referree) AS web_count
    FROM relation
    GROUP BY referrer
),
common AS (
    SELECT a.referrer AS web1, b.referrer AS web2, COUNT(*) AS
com_cnt
    FROM relation a
    JOIN relation b ON a.referree = b.referree
    WHERE a.referrer < b.referrer
    GROUP BY a.referrer, b.referrer
)
SELECT common.web1, common.web2,
CASE WHEN r1.web_count + r2.web_count - common.com_cnt > 0
THEN common.com_cnt / (r1.web_count + r2.web_count - common.com_cnt)
ELSE 0 END AS similarity

```

```

FROM common JOIN (
    SELECT referrer, web_count
    FROM ref_count
    WHERE referrer IN (SELECT web1 FROM common)
) r1 ON common.web1 = r1.referrer
JOIN (
    SELECT referrer, web_count
    FROM ref_count
    WHERE referrer IN (SELECT web2 FROM common)
) r2 ON common.web2 = r2.referrer order by similarity desc limit 500;

```

查询结果:

```

26531267      89894179      1.0
39381817      77016942      1.0
25990595      27394033      1.0
42080746      89894179      1.0
127989625     412486871     1.0
138239358     159570745     1.0
16804125      128953441     1.0
14314752      21093061      1.0
12742918      285949147     1.0
40296131      437136030     1.0
27889544      86525478      1.0
40439102      97742874      1.0
31136356      34054226      1.0
20490152      29944159      1.0
326386597     437136030     1.0
60394221      105639498     1.0
13610348      48739090      1.0
14760311      52659511      1.0
121098486     427194018     1.0
15792401      21093061      1.0
171803433     345999006     1.0
31493102      52659511      1.0
15387913      86376769      1.0
92609504      345999006     1.0
237286932     285949147     1.0
31652107      48259437      1.0
14592440      437136030     1.0
20597466      80863675      1.0
Time taken: 202.238 seconds, Fetched 500 row(s)
25/02/25 14:44:47 INFO CliDriver: Time taken: 202.238 seconds, Fetched 500 row(s)
spark-sql> |

```

耗时 202.238s 完成

- intersect 操作:

查询语句:

```

WITH ref_count AS (
    SELECT referrer, COUNT(DISTINCT referree) AS web_count
    FROM relation GROUP BY referrer),
common as (select a.referrer as web1, b.referrer as web2, count(*)
as com_cnt
    from relation a join relation b
    on a.referree = b.referree
    where a.referrer < b.referrer
    group by a.referrer, b.referrer)
SELECT common.web1, common.web2,
CASE WHEN r1.web_count + r2.web_count - common.com_cnt > 0
THEN common.com_cnt / (r1.web_count + r2.web_count - common.com_cnt)
ELSE 0 END AS similarity
FROM common JOIN (
    SELECT referrer, web_count FROM ref_count

```

```

WHERE referrer IN (
    SELECT web1 FROM common
    INTERSECT
    SELECT referrer FROM ref_count)
) r1 ON common.web1 = r1.referrer
JOIN (
    SELECT referrer, web_count
    FROM ref_count
    WHERE referrer IN (
        SELECT web2 FROM common
        INTERSECT
        SELECT referrer FROM ref_count)
) r2 ON common.web2 = r2.referrer order by similarity desc limit 500;

```

查询结果:

```

39381817      77616942      1.0
25990595      27394033      1.0
29152253      89094179      1.0
344454081     412486071     1.0
25500358      159570745     1.0
16804125      128953441     1.0
15792401      21093061      1.0
12742918      285949147     1.0
40296131      437136030     1.0
27889544      86525478      1.0
17864133      97742874      1.0
33845640      34054226      1.0
20498152      29944159      1.0
33129732      437136030     1.0
93606448      105639498     1.0
13610348      48739090      1.0
27125224      52659511      1.0
121098486     427194018     1.0
16728776      21093061      1.0
171003433     345999006     1.0
31493102      52659511      1.0
15387913      86376769      1.0
92609504      345999006     1.0
40125679      285949147     1.0
31652187      48259437      1.0
14592440      437136030     1.0
14637838      22299596      1.0
Time taken: 205.164 seconds, Fetched 500 row(s)
25/02/25 14:56:56 INFO CliDriver: Time taken: 205.164 seconds, Fetched 500 row(s)
spark-sql> |

```

耗时 205.164s 完成

- exists 方式:

查询语句:

```

WITH ref_count AS (
    SELECT referrer, COUNT(DISTINCT referree) AS web_count
    FROM relation
    GROUP BY referrer),
common AS (
    SELECT a.referrer AS web1, b.referrer AS web2, COUNT(*) AS
com_cnt
    FROM relation a
    JOIN relation b ON a.referree = b.referree
    WHERE a.referrer < b.referrer
    GROUP BY a.referrer, b.referrer)

```

```

SELECT
    common.web1,
    common.web2,
    CASE
        WHEN r1.web_count + r2.web_count - common.com_cnt > 0
        THEN common.com_cnt / (r1.web_count + r2.web_count -
common.com_cnt)
        ELSE 0
    END AS similarity
FROM common JOIN (
    SELECT referrer, web_count
    FROM ref_count r
    WHERE EXISTS (SELECT 1 FROM common c WHERE c.web1 = r.referrer)
) r1 ON common.web1 = r1.referrer
JOIN (
    SELECT referrer, web_count
    FROM ref_count r
    WHERE EXISTS (SELECT 1 FROM common c WHERE c.web2 = r.referrer)
) r2 ON common.web2 = r2.referrer
ORDER BY similarity DESC LIMIT 500;

```

查询结果:

```

69678717      89894179      1.0
39381817      77816942      1.0
26036977 22817535      1.0
33749188      89894179      1.0
14193736      15487600      1.0
18877266      41866950      1.0
16804125      128953441     1.0
14791225      21893061      1.0
14137741      17550904      1.0
14592440      437136030     1.0
27889544      86525478      1.0
27945434      97742874      1.0
28828804      34854226      1.0
21699833      29944159      1.0
378461352     437136030     1.0
19158800      105639498     1.0
8291188 17219913      1.0
27125224      52659511      1.0
121898486     427194818     1.0
7953008 21093061      1.0
337414203     345999066     1.0
27377864      52659511      1.0
15387913      86376769      1.0
92609584      345999066     1.0
5723618 41866950      1.0
31652187      48259437      1.0
22084402      437136030     1.0
14637838      22299596      1.0
Time taken: 204.174 seconds, Fetched 500 row(s)
25/02/25 14:52:40 INFO CliDriver: Time taken: 204.174 seconds, Fetched 500 row(s)
spark-sql> |

```

耗时 204.174s 完成。

五、 查询验证

为了方便后续查询验证,我先前将临时表 ref_count,common 保存为了中间表 tmp_ref_count,tmp_common,相似度计算结果表保存到了 result。

```

25/02/25 14:57:59 INFO scheduler.DAGScheduler: Job 42 finished: processCmd at CliDriver.java:376, took 0.021891 s
web_large      relation      false
web_large      result      false
web_large      tmp_common  false
web_large      tmp_ref_count false
Time taken: 0.036 seconds, Fetched 4 row(s)
25/02/25 14:57:59 INFO CliDriver: Time taken: 0.036 seconds, Fetched 4 row(s)
spark-sql> select count(*) from result;

```


查询 result 表数据行数:

```
25/02/25 14:59:20 INFO scheduler.DAGScheduler: Job 43 finished: processCmd at CliDriver.java:376, took 6.931805 s
64195979
Time taken: 7.678 seconds, Fetched 1 row(s)
25/02/25 14:59:20 INFO CliDriver: Time taken: 7.678 seconds, Fetched 1 row(s)
spark-sql> |
```

耗时 7678s 完成, 显示结果 64195979 行, 可见表及其庞大。

1. 查询相似度位于 0.72~0.73 的记录, 返回最多 30 条。

```
Time taken: 19.932 seconds, Fetched 30 row(s)
25/02/21 21:55:42 INFO CliDriver: Time taken: 19.932 seconds, Fetched 30
row(s)
spark-sql> select * from result where similarity between 0.72 and 0.73 li
mit 30;|
```

```
25/02/21 21:57:46 INFO scheduler.TaskSetManager: Finished task 99.0 in st
age 317.0 (TID 31346) in 227 ms on localhost (executor driver) (100/100)
25/02/21 21:57:46 INFO scheduler.TaskSchedulerImpl: Removed TaskSet 317.0
, whose tasks have all completed, from pool
25/02/21 21:57:46 INFO scheduler.DAGScheduler: ResultStage 317 (processCm
d at CliDriver.java:376) finished in 15.950 s
25/02/21 21:57:46 INFO scheduler.DAGScheduler: Job 138 finished: processC
md at CliDriver.java:376, took 15.953909 s
291245333      296171249      0.7297297297297297
25227543      334462096      0.7297297297297297
99786878      122113661      0.7297297297297297
28539307      437057716      0.7297297297297297
15594319      29725736       0.7297297297297297
14667720      500021610      0.7297297297297297
42098840      148230772      0.7297297297297297
19040457      46970699       0.7297297297297297
131785632     202186645      0.7297297297297297
24033843      64058908       0.7297297297297297
18088315      28393099       0.7297297297297297
16987309      30211029       0.7295918367346939
17677642      18559386       0.7295597484276729
57046183      69306513       0.7295081967213115
16809676      16929288       0.7295081967213115
79378664      90433572       0.7295081967213115
15594319      35415483       0.7294117647058823
18359358      19040457       0.7294117647058823
291245333     370352656      0.7294117647058823
286481558     338614684      0.7294117647058823
16541224      260975690      0.7294117647058823
18359358      177761713      0.7294117647058823
128496409     273186922      0.7294117647058823
395472459     523832662      0.7294117647058823
1976847 45869579 0.7293233082706767
1976847 17023350 0.7293233082706767
1976847 16956336 0.7293233082706767
294854141     368829932      0.7292817679558011
397464137     554003477      0.7292817679558011
327129 9659998 0.7291666666666666
Time taken: 19.945 seconds, Fetched 30 row(s)
25/02/21 21:57:46 INFO CliDriver: Time taken: 19.945 seconds, Fetched 30
row(s)
spark-sql> |
```

耗时 19.945s 完成查询, 返回了 30 行数据。

2. 查询相似度为 0.9 的记录, 最多 30 条。

```
25/02/21 21:57:46 INFO CliDriver: Time taken: 19.945 seconds, Fetched 30
row(s)
spark-sql> select * from result where similarity=0.9 limit 30;|
```



```

25/02/21 21:59:44 INFO scheduler.TaskSetManager: Finished task 74.0 in stage 322.0 (TID 31546) in 4 ms on localhost (executor driver) (75/75)
25/02/21 21:59:44 INFO scheduler.TaskSchedulerImpl: Removed TaskSet 322.0, whose tasks have all completed, from pool
25/02/21 21:59:44 INFO scheduler.DAGScheduler: ResultStage 322 (processCmd at CliDriver.java:376) finished in 12.442 s
25/02/21 21:59:44 INFO scheduler.DAGScheduler: Job 143 finished: processCmd at CliDriver.java:376, took 12.444734 s
17004145      21764349      0.9
116225517    123054765      0.9
816000 16955876      0.9
20216090    21624408      0.9
17004145    19662636      0.9
543643490   543723381      0.9
395029895   399644865      0.9
543633230   548738861      0.9
7265488 18324692      0.9
19424894    40977581      0.9
14880365    14882461      0.9
18324692    40977581      0.9
14799157    218588666      0.9
14603521    16929565      0.9
547878710   548738861      0.9
21624408    105508695      0.9
14268799    18324692      0.9
14268799    19424894      0.9
18324692    97308937      0.9
816000 14603521      0.9
9557358 19424894      0.9
16929565    16955876      0.9
21624408    45922281      0.9
16956014    17001049      0.9
Time taken: 32.389 seconds, Fetched 24 row(s)
25/02/21 21:59:44 INFO CliDriver: Time taken: 32.389 seconds, Fetched 24 row(s)
spark-sql> |

```

耗时 32.389s 完成查询，返回了 24 项相似度为 0.9 的数据。

至此，实验完成。

六、 实验总结

本次实验基于 Hadoop HDFS 和 Spark SQL 构建分布式数据库查询系统，我深入理解了大数据存储与处理的分布式架构。通过实验，掌握了 HDFS 的文件存储机制及 Spark SQL 的分布式查询优化技术，体验了其在海量数据下的高效查询性能。实验过程中，数据分区、并行计算和内存优化等策略显著提升了查询效率，同时也认识到集群资源调度与数据倾斜问题的挑战。总体而言，我熟悉了手动搭建 hadoop 和 spark-sql 的基本流程，学会了对大数据集查询的 sql 语句进行一定程度优化，本学期的数据库实验让我受益匪浅！