

transaction: 实体关系抽取实验

一、先验数据导入

进入 transaction 文件夹后，修改用户权限

```
frankstein@ubuntu:~/CNdeepdive/CNdeepdive/transaction$ chmod -R 777 ./
```

创建数据库，并指定所有者

```
postgres=# CREATE DATABASE trans_test OWNER ke;
CREATE DATABASE
postgres=# \q
postgres@ubuntu:~$ psql -U ke -d trans_test -h 127.0.0.1 -p 5432
psql (9.5.25)
SSL connection (protocol: TLSv1.2, cipher: ECDHE-RSA-AES256-GCM-SHA384, bits: 256, compression: off)
Type "help" for help.

trans_test=> █
```

在本地 postgresql 中为项目建立数据库，再在项目文件夹下建立数据库配置文件生成 postgresql 数据表

```
frankstein@ubuntu:~/CNdeepdive/CNdeepdive/transaction$ echo "postgresql://ke@localhost:5432/trans_test">db.url
frankstein@ubuntu:~/CNdeepdive/CNdeepdive/transaction$ deepdive compile && deepdive do transaction_dbdata
2025-05-30 04:11:16.540159 'run/LATEST.COMPILE' -> '20250530/041116.315793233'
2025-05-30 04:11:16.540771 'run/RUNNING.COMPILE' -> '20250530/041116.315793233'
2025-05-30 04:11:16.540806 Parsing DeepDive application (/home/frankstein/CNdeepdive/CNdeepdive/transaction) to generate:
2025-05-30 04:11:16.540823 run/compiled/schema.json
2025-05-30 04:11:16.540833 run/compiled/schema.json
2025-05-30 04:11:47.348977 # Done: N/A
2025-05-30 04:11:47.349000 # no-op
2025-05-30 04:11:47.349022 mark_done data/transaction_dbdata
2025-05-30 04:11:47.374447 #####
#####
2025-05-30 04:11:47.374619
2025-05-30 04:11:47.374669
'run/FINISHED' -> '20250530/041123.827399787'
frankstein@ubuntu:~/CNdeepdive/CNdeepdive/transaction$ █
```

二、待抽取文章导入

导入文章到 postgresql 中

```

frankstein@ubuntu:~/CNdeepdive/CNdeepdive/transaction$ deepdive do articles
'run/RUNNING' -> '20250530/041355.511407471'
'run/LATEST' -> '20250530/041355.511407471'
2025-05-30 04:14:03.080485 # on ubuntu: deepdive do articles
2025-05-30 04:14:03.080622 # run/20250530/041355.511407471/plan.sh
2025-05-30 04:14:03.080652 # execution plan for data/articles
2025-05-30 04:14:03.080798
2025-05-30 04:14:03.080824 : ## process/init/app #####
#####
2025-05-30 04:14:03.080845 : # Done: 2025-05-30T04:11:46-0700 (2m 9s ago)
2025-05-30 04:14:03.080856 : process/init/app/run.sh
2025-05-30 04:14:03.080865 : mark_done process/init/app
2025-05-30 04:14:03.080875 : #####
#####
2025-05-30 04:14:03.081040
2025-05-30 04:14:03.081094
2025-05-30 04:14:03.081106 ## process/init/relation/articles #####
#####
2025-05-30 04:14:03.081116 # Done: N/A
2025-05-30 04:14:03.081126 process/init/relation/articles/run.sh
2025-05-30 04:14:03.081135 ++ dirname process/init/relation/articles/run.sh
2025-05-30 04:14:03.081276 + cd process/init/relation/articles
2025-05-30 04:14:03.081341 + export DEEPDIVE_CURRENT_PROCESS_NAME=process/init/relation/ar
ticles
2025-05-30 04:14:03.081418 + DEEPDIVE_CURRENT_PROCESS_NAME=process/init/relation/articles
2025-05-30 04:14:03.081430 + deepdive create table articles
2025-05-30 04:14:03.274542 NOTICE: table "articles" does not exist, skipping
2025-05-30 04:14:03.281418 CREATE TABLE
2025-05-30 04:14:03.286391 + deepdive load articles
loading articles: 0:00:00 343KiB [ 2.8MiB/s] ([ 2.8MiB/s])cles.csv (csv format)
loading articles: 0:00:00 50 [ 411 /s] ([ 411 /s])
2025-05-30 04:14:03.598123 COPY 50
2025-05-30 04:14:03.619302 mark_done process/init/relation/articles
2025-05-30 04:14:03.640898 #####
#####
2025-05-30 04:14:03.641095
2025-05-30 04:14:03.641138
2025-05-30 04:14:03.641164 ## data/articles #####
#####
2025-05-30 04:14:03.641186 # Done: N/A
2025-05-30 04:14:03.641207 # no-op
2025-05-30 04:14:03.641228 mark_done data/articles
2025-05-30 04:14:03.658438 #####
#####
2025-05-30 04:14:03.658562
2025-05-30 04:14:03.658590
'run/FINISHED' -> 'run/FINISHED~'
'run/FINISHED' -> '20250530/041355.511407471'
frankstein@ubuntu:~/CNdeepdive/CNdeepdive/transaction$

```

这里减少行数，减少 articles 的行数来缩短后面程序运行时间。将 50 行减少为 5 行。
重新导入文章到 postgresql 中

```

frankstein@ubuntu:~/CNdeepdive/CNdeepdive/transaction$ cd input
frankstein@ubuntu:~/CNdeepdive/CNdeepdive/transaction/input$ ls
articlesall.csv init.sh total.parsed transaction_dbdata.csv
frankstein@ubuntu:~/CNdeepdive/CNdeepdive/transaction/input$ head -n 5 articlesall.csv >ar
ticles.csv
frankstein@ubuntu:~/CNdeepdive/CNdeepdive/transaction/input$ cd ..
frankstein@ubuntu:~/CNdeepdive/CNdeepdive/transaction$ deepdive sql
psql (9.5.25)
SSL connection (protocol: TLSv1.2, cipher: ECDHE-RSA-AES256-GCM-SHA384, bits: 256, compres
sion: off)
Type "help" for help.

trans_test=> TRUNCATE articles;
TRUNCATE TABLE
trans_test=> \q
frankstein@ubuntu:~/CNdeepdive/CNdeepdive/transaction$ deepdive do articles
All done at 2025-05-30T05:40:58-0700 (6m 2s ago):
articles
To see more details, use: deepdive plan
To redo them anyway, use: deepdive redo
frankstein@ubuntu:~/CNdeepdive/CNdeepdive/transaction$ deepdive redo articles
frankstein@ubuntu:~/CNdeepdive/CNdeepdive/transaction$
#####
2025-05-30 05:47:26.751021
2025-05-30 05:47:26.751051
'run/FINISHED' -> 'run/FINISHED~'
'run/FINISHED' -> '20250530/054723.655353569'

```

deepdive 可以通过 query 语句查询数据库数据。查询 id 检验导入是否成功：


```
frankstein@ubuntu:~/CNdeepdive/CNdeepdive/transaction$ deepdive query '?- articles(id, _).
id
-----
1201734370
1201734454
1201734455
1201734457
1201734458
(5 rows)
```

三、用 nlp 模块进行文本处理

进入 bazaar/parser 目录下，执行编译命令，编译完成后会在 target 中生成可执行文件

```
frankstein@ubuntu:~/CNdeepdive/CNdeepdive/transaction/udf/bazaar/parser$ sbt/sbt stage
Getting org.scala-sbt sbt 0.13.8 ...
:: retrieving :: org.scala-sbt#boot-app
  confs: [default]
  52 artifacts copied, 0 already retrieved (17674kB/246ms)
Getting Scala 2.10.4 (for sbt)...
:: retrieving :: org.scala-sbt#boot-scala
  confs: [default]
  5 artifacts copied, 0 already retrieved (24459kB/197ms)
[info] Loading project definition from /home/frankstein/CNdeepdive/CNdeepdive/transaction/
udf/bazaar/parser/project
[info] Updating {file:/home/frankstein/CNdeepdive/CNdeepdive/transaction/udf/bazaar/parser
/project/}parser-build...
[info] Resolving org.scala-sbt.ivy#ivy;2.3.0-sbt-fccfbd44c9f64523b61398a0155784dcbbaeae28f
[info] Resolving org.fusesource.jansi#jansi;1.4 ...
[info] Done updating.
[info] Set current project to deepdive-nlp-parser (in build file:/home/frankstein/CNdeepdi
ve/CNdeepdive/transaction/udf/bazaar/parser/)
[info] Updating {file:/home/frankstein/CNdeepdive/CNdeepdive/transaction/udf/bazaar/parser
/}parser...
[info] Resolving org.fusesource.jansi#jansi;1.4 ...
[info] Done updating.
[warn] Scala version was updated by one of library dependencies:
[warn] * org.scala-lang:scala-library:(2.10.3, 2.10.4, 2.10.2, 2.10.0) -> 2.10.5
[warn] To force scalaVersion, add the following:
[warn] ivyScala := ivyScala.value map { _.copy(overrideScalaVersion = true) }
[warn] Run 'evicted' to see detailed eviction warnings
[info] Compiling 6 Scala sources to /home/frankstein/CNdeepdive/CNdeepdive/transaction/udf
/bazaar/parser/target/scala-2.10/classes...
[info] Wrote start script for mainClass := Some(com.clearcut.nlp.Main) to /home/frankstein
/CNdeepdive/CNdeepdive/transaction/udf/bazaar/parser/target/start
[success] Total time: 23 s, completed May 30, 2025 4:20:25 AM
frankstein@ubuntu:~/CNdeepdive/CNdeepdive/transaction/udf/bazaar/parser$
```

编译并执行生成 sentences 数据表

```
frankstein@ubuntu:~/CNdeepdive/CNdeepdive/transaction/udf/bazaar/parser$ deepdive compile
&& deepdive do sentences
2025-05-30 06:31:22.103309 'run/LATEST.COMPILE' -> '20250530/063122.003964056'
2025-05-30 06:31:22.103682 'run/RUNNING.COMPILE' -> '20250530/063122.003964056'
2025-05-30 06:31:22.103740 Parsing DeepDive application (/home/frankstein/CNdeepdive/CNdee
pdive/transaction) to generate:
2025-05-30 06:31:22.103789 run/compiled/schema.json
2025-05-30 06:31:22.103835 from app.ddlog
2025-05-30 06:31:23.607284 run/compiled/deepdive.conf
2025-05-30 06:31:23.607410 from app.ddlog
2025-05-30 06:31:25.252737 from deepdive.conf
2025-05-30 06:31:25.259320 run/compiled/deepdive.conf.json
2025-05-30 06:31:25.980833 Performing sanity checks on run/compiled/deepdive.conf.json:
2025-05-30 06:31:26.053893 checking if input_extractors_well_defined
2025-05-30 06:31:26.053977 checking if input_schema_wellformed
2025-05-30 06:31:26.054198 Normalizing and adding built-in processes to the data flow to c
ompile:
2025-05-30 06:31:26.054518 run/compiled/config-0.00-init objects.json
```



```

frankstein@ubuntu:~/CNdeepdive/CNdeepdive/transaction/udf/bazaar/parser$ deepdive compile
&& deepdive do company_mention
2025-05-30 07:50:38.930480 'run/LATEST.COMPILE' -> '20250530/075038.858538317'
2025-05-30 07:50:38.930644 'run/RUNNING.COMPILE' -> '20250530/075038.858538317'
2025-05-30 07:50:38.930863 Parsing DeepDive application (/home/frankstein/CNdeepdive/CNdeepdive/transaction) to generate:
2025-05-30 07:50:38.930985 run/compiled/schema.json
2025-05-30 07:50:38.931108 from app.ddlog
2025-05-30 07:50:40.532039 run/compiled/deepdive.conf
2025-05-30 07:50:40.533646 from app.ddlog
2025-05-30 07:50:42.365161 from deepdive.conf
2025-05-30 07:50:42.370356 run/compiled/deepdive.conf.json
2025-05-30 07:50:43.038299 Performing sanity checks on run/compiled/deepdive.conf.json:
2025-05-30 07:50:43.120888 checking if input_extractors_well_defined
2025-05-30 07:50:43.121370 checking if input_schema_wellformed
2025-05-30 07:50:43.121408 Normalizing and adding built-in processes to the data flow to compile:

```

```

2025-05-30 07:53:46.116758 # Done: 2025-05-30T07:48:20-0700 (5m 18s ago)
2025-05-30 07:53:46.116777 # no-op
2025-05-30 07:53:46.116796 mark_done data/company_mention
2025-05-30 07:53:46.136354 #####
#####
2025-05-30 07:53:46.136481
2025-05-30 07:53:46.136510
'run/FINISHED' -> 'run/FINISHED~'
'run/FINISHED' -> '20250530/075338.135863963'

```

编译并执行，生成候选实体表

```

frankstein@ubuntu:~/CNdeepdive/CNdeepdive/transaction/udf/bazaar/parser$ deepdive compile
&& deepdive do transaction_candidate
2025-05-30 07:48:06.748243 'run/LATEST.COMPILE' -> '20250530/074806.686577747'
2025-05-30 07:48:06.748321 'run/RUNNING.COMPILE' -> '20250530/074806.686577747'
2025-05-30 07:48:06.748352 Parsing DeepDive application (/home/frankstein/CNdeepdive/CNdeepdive/transaction) to generate:
2025-05-30 07:48:06.748363 run/compiled/schema.json
2025-05-30 07:48:06.748373 from app.ddlog
2025-05-30 07:48:08.517964 run/compiled/deepdive.conf
2025-05-30 07:48:08.518090 from app.ddlog
2025-05-30 07:48:10.103022 from deepdive.conf
2025-05-30 07:48:10.107636 run/compiled/deepdive.conf.json
2025-05-30 07:48:10.705798 Performing sanity checks on run/compiled/deepdive.conf.json:
2025-05-30 07:48:10.806641 checking if input_extractors_well_defined
2025-05-30 07:48:10.806786 checking if input_schema_wellformed
2025-05-30 07:48:10.806806 Normalizing and adding built-in processes to the data flow to compile:

```

```

2025-05-30 07:48:23.486432 mark_done process/ext_transaction_candidate_by_map_transaction_candidate
2025-05-30 07:48:23.512798 #####
#####
2025-05-30 07:48:23.512911
2025-05-30 07:48:23.512930
2025-05-30 07:48:23.512944 ## data/transaction_candidate #####
#####
2025-05-30 07:48:23.512958 # Done: N/A
2025-05-30 07:48:23.512972 # no-op
2025-05-30 07:48:23.512986 mark_done data/transaction_candidate
2025-05-30 07:48:23.526660 #####
#####
2025-05-30 07:48:23.526790
2025-05-30 07:48:23.526819
'run/FINISHED' -> 'run/FINISHED~'
'run/FINISHED' -> '20250530/074813.579134331'

```

五、特征提取

编译并执行，生成特征数据库

```

frankstein@ubuntu:~/CNdeepdive/CNdeepdive/transaction/udf/bazaar/parser$ deepdive compile
&& deepdive do transaction_feature
2025-05-30 07:55:22.133692 'run/LATEST.COMPILE' -> '20250530/075522.064909982'
2025-05-30 07:55:22.133795 'run/RUNNING.COMPILE' -> '20250530/075522.064909982'
2025-05-30 07:55:22.133813 Parsing DeepDive application (/home/frankstein/CNdeepdive/CNdeepdive/transaction) to generate:
2025-05-30 07:55:22.133922 run/compiled/schema.json
2025-05-30 07:55:22.133983 from app.ddlog
2025-05-30 07:55:23.584576 run/compiled/deepdive.conf
2025-05-30 07:55:23.584681 from app.ddlog
2025-05-30 07:55:25.434818 from deepdive.conf
2025-05-30 07:55:25.440682 run/compiled/deepdive.conf.json
2025-05-30 07:55:26.091575 Performing sanity checks on run/compiled/deepdive.conf.json:
2025-05-30 07:55:26.182897 checking if input_extractors_well_defined
2025-05-30 07:55:26.183136 checking if input_schema_wellformed
2025-05-30 07:55:26.183678 Normalizing and adding built-in processes to the data flow to compile:

2025-05-30 07:55:35.178174 # Done: N/A
2025-05-30 07:55:35.178196 # no-op
2025-05-30 07:55:35.178218 mark_done data/transaction_feature
2025-05-30 07:55:35.199131 #####
#####
2025-05-30 07:55:35.199246
2025-05-30 07:55:35.199276
'run/FINISHED' -> 'run/FINISHED~'
'run/FINISHED' -> '20250530/075528.859578969'

```

执行如下语句，查看生成结果

```

frankstein@ubuntu:~/CNdeepdive/CNdeepdive/transaction/udf/bazaar/parser$ deepdive query '
20 ?- transaction_feature(_,_, feature).'
feature
-----
IS_INVERTED
INV_WORD_SEQ [郴州市 城市 建设 投资 发展 集团 有限 公司]
INV_LEMMA_SEQ [郴州市 城市 建设 投资 发展 集团 有限 公司]
INV_NER_SEQ [ORG ORG ORG ORG ORG ORG ORG ORG]
INV_POS_SEQ [NR NN NN NN NN NN NN JJ NN]
INV_W_LEMMA_L_1_R_1 [为] [提供]
INV_W_NER_L_1_R_1 [0] [0]
INV_W_LEMMA_L_1_R_2 [为] [提供 担保]
INV_W_NER_L_1_R_2 [0] [0 0]
INV_W_LEMMA_L_1_R_3 [为] [提供 担保 公告]
INV_W_NER_L_1_R_3 [0] [0 0 0]
INV_W_LEMMA_L_2_R_1 [公司 为] [提供]
INV_W_NER_L_2_R_1 [ORG 0] [0]
INV_W_LEMMA_L_2_R_2 [公司 为] [提供 担保]
INV_W_NER_L_2_R_2 [ORG 0] [0 0]
INV_W_LEMMA_L_2_R_3 [公司 为] [提供 担保 公告]
INV_W_NER_L_2_R_3 [ORG 0] [0 0 0]
INV_W_LEMMA_L_3_R_1 [有限 公司 为] [提供]
INV_W_NER_L_3_R_1 [ORG ORG 0] [0]
INV_W_LEMMA_L_3_R_2 [有限 公司 为] [提供 担保]
(20 rows)

```

六、样本打标

执行以下命令，得到最终标签。


```

frankstein@ubuntu:~/CNdeepdive/CNdeepdive/transaction/udf/bazaar/parser$ deepdive do trans
action_label_resolved
'run/RUNNING' -> '20250530/080411.247836758'
'run/LATEST' -> '20250530/080411.247836758'
2025-05-30 08:04:13.774388 # on ubuntu: deepdive do transaction_label_resolved
2025-05-30 08:04:13.774864 # run/20250530/080411.247836758/plan.sh
2025-05-30 08:04:13.775181 # execution plan for data/transaction_label_resolved
2025-05-30 08:04:13.775397
2025-05-30 08:04:13.775620 : ## process/init/app #####
#####
2025-05-30 08:04:13.775835 : # Done: 2025-05-30T04:11:46-0700 (3h 52m 25s ago)
2025-05-30 08:04:13.776043 : process/init/app/run.sh
2025-05-30 08:04:13.776250 : mark_done process/init/app
2025-05-30 08:04:13.776460 : #####
#####
2025-05-30 08:04:13.776673
2025-05-30 08:04:13.776879
2025-05-30 08:04:13.777085 : ## process/init/relation/articles #####
#####
2025-05-30 08:04:13.777320 : # Done: 2025-05-30T05:47:26-0700 (2h 16m 45s ago)
2025-05-30 08:04:13.777536 : process/init/relation/articles/run.sh
2025-05-30 08:04:13.777739 : mark_done process/init/relation/articles
2025-05-30 08:04:13.777945 : #####
#####
2025-05-30 08:04:15.531354
2025-05-30 08:04:15.531382
2025-05-30 08:04:15.531408 ## data/transaction_label_resolved #####
#####
2025-05-30 08:04:15.531429 # Done: N/A
2025-05-30 08:04:15.531451 # no-op
2025-05-30 08:04:15.531472 mark_done data/transaction_label_resolved
2025-05-30 08:04:15.547678 #####
#####
2025-05-30 08:04:15.547853
2025-05-30 08:04:15.547902
'run/FINISHED' -> 'run/FINISHED~'
'run/FINISHED' -> '20250530/080411.247836758'

```

七、模型构建

最后编译执行决策表:

```

frankstein@ubuntu:~/CNdeepdive/CNdeepdive/transaction/udf/bazaar/parser$ deepdive compile
&& deepdive do has_transaction
2025-05-30 08:05:46.209887 'run/LATEST.COMPILE' -> '20250530/080546.148916208'
2025-05-30 08:05:46.210022 'run/RUNNING.COMPILE' -> '20250530/080546.148916208'
2025-05-30 08:05:46.210088 Parsing DeepDive application (/home/frankstein/CNdeepdive/CNdee
pdive/transaction) to generate:
2025-05-30 08:05:46.210116 run/compiled/schema.json
2025-05-30 08:05:46.210139 from app.ddlog
2025-05-30 08:05:47.460913 run/compiled/deepdive.conf
2025-05-30 08:05:47.461015 from app.ddlog
2025-05-30 08:05:49.174786 from deepdive.conf
2025-05-30 08:05:49.180462 run/compiled/deepdive.conf.json
2025-05-30 08:05:49.792939 Performing sanity checks on run/compiled/deepdive.conf.json:
2025-05-30 08:05:49.870510 checking if input_extractors_well_defined
2025-05-30 08:05:49.870636 checking if input_schema_wellformed
2025-05-30 08:05:49.870664 Normalizing and adding built-in processes to the data flow to c
ompile:

```

```

2025-05-30 08:05:58.401919 mark_done process/ext_has_transaction
2025-05-30 08:05:58.411487 #####
#####
2025-05-30 08:05:58.411558
2025-05-30 08:05:58.411573
2025-05-30 08:05:58.411585 ## data/has_transaction #####
#####
2025-05-30 08:05:58.411595 # Done: N/A
2025-05-30 08:05:58.411606 # no-op
2025-05-30 08:05:58.411616 mark_done data/has_transaction
2025-05-30 08:05:58.423272 #####
#####
2025-05-30 08:05:58.423393
2025-05-30 08:05:58.423423
'run/FINISHED' -> 'run/FINISHED~'
'run/FINISHED' -> '20250530/080552.854433040'

```

编译并生成最终的概率模型

```

frankstein@ubuntu:~/CNdeepdive/CNdeepdive/transaction/udf/bazaar/parser$ deepdive compile
&& deepdive do probabilities
2025-05-30 08:08:19.654300 'run/LATEST.COMPILE' -> '20250530/080819.573053249'
2025-05-30 08:08:19.654597 'run/RUNNING.COMPILE' -> '20250530/080819.573053249'
2025-05-30 08:08:19.654706 Parsing DeepDive application (/home/frankstein/CNdeepdive/CNdee
pdive/transaction) to generate:
2025-05-30 08:08:19.654806 run/compiled/schema.json
2025-05-30 08:08:19.654901 from app.ddlog
2025-05-30 08:08:21.276465 run/compiled/deepdive.conf
2025-05-30 08:08:21.276628 from app.ddlog
2025-05-30 08:08:23.021128 from deepdive.conf
2025-05-30 08:08:23.028231 run/compiled/deepdive.conf.json
2025-05-30 08:08:23.691441 Performing sanity checks on run/compiled/deepdive.conf.json:
2025-05-30 08:08:23.777712 checking if input_extractors_well_defined
2025-05-30 08:08:23.777833 checking if input_schema_wellformed
2025-05-30 08:08:23.778980 Normalizing and adding built-in processes to the data flow to c
ompile:
2025-05-30 08:09:36.511525 mark_done process/model/load_probabilities
2025-05-30 08:09:36.523265 #####
#####
2025-05-30 08:09:36.523562
2025-05-30 08:09:36.524120
2025-05-30 08:09:36.524140 ## data/model/probabilities #####
#####
2025-05-30 08:09:36.524151 # Done: N/A
2025-05-30 08:09:36.524161 # no-op
2025-05-30 08:09:36.524170 mark_done data/model/probabilities
2025-05-30 08:09:36.551672 #####
#####
2025-05-30 08:09:36.551811
2025-05-30 08:09:36.551838
'run/FINISHED' -> 'run/FINISHED~'
'run/FINISHED' -> '20250530/080826.735798012'

```

查看预测的公司间交易关系概率。

这里因为我之前将 50 行减少为 5 行，座椅这里查出来的数据只有 16 行


```
frankstein@ubuntu:~/CNdeepdive/CNdeepdive/transaction/udf/bazaar/parser$ deepdive sql "SELECT p1_id,p2_id,expectation FROM has_transaction_label_inference ORDER BY random() LIMIT 20"
```

p1_id	p2_id	expectation
1201734457_18_7_11	1201734457_18_13_18	0.921
1201734457_3_5_10	1201734457_3_64_68	0.001
1201734457_3_46_51	1201734457_3_64_68	0.014
1201734457_3_64_68	1201734457_3_46_51	0.022
1201734457_3_22_27	1201734457_3_46_51	0.071
1201734457_3_22_27	1201734457_3_5_10	0.094
1201734457_3_64_68	1201734457_3_5_10	0.001
1201734457_24_28_29	1201734457_24_19_20	0.412
1201734457_3_22_27	1201734457_3_64_68	0.003
1201734457_3_46_51	1201734457_3_22_27	0.072
1201734457_3_5_10	1201734457_3_46_51	0.013
1201734457_3_46_51	1201734457_3_5_10	0.014
1201734457_18_13_18	1201734457_18_7_11	0.915
1201734457_3_64_68	1201734457_3_22_27	0.005
1201734457_3_5_10	1201734457_3_22_27	0.095
1201734457_24_19_20	1201734457_24_28_29	0.415

(16 rows)