


Неструктурирани бази на податоци	Граф-податоци
	Група: -Виктор Јовев (186068) -Марко Марковиќ (181507) -Тамара Малинова (185061) -Никола Петроски(181501) -Леона Илиева (181018)

Податочно множество: <https://snap.stanford.edu/data/ego-Gplus.html>

Датотеки:

nodeId.edges : Врските во его-мрежата за јазелот nodeId. Врските се насочени за множеството google+. Его-јазелот не се појавува, но се претпоставува дека тие го следат секој јазел кој се појавува во датотеката.

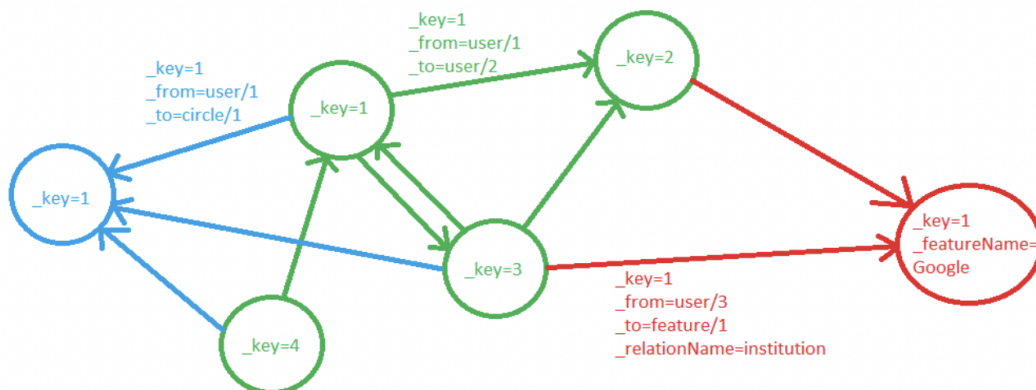
nodeId.circles : Множество од кругови за секој его-јазел. Секоја линија содржи по еден круг заедно со серија од идентификациони јазли. Првиот токен од секоја линија е името на кругот.

nodeId.feats : Карактеристиките за секој од јазлите кои се појавуваат во датотеката
nodeId.edges

nodeId.egofeat : Карактеристиките на секој его-јазел

nodeId.featsnames : Прозивот за секоја карактеристична димензија. Карактеристиките означени со "1" се доколку корисникот го има својството за нивниот профил, инаку "0".

Граф модел



Neo4j потребно време за импорт: 25 минути

Првично, суровите податоци беа обработени и зачувани во .csv датотеки со цел полесен импорт на податоците во базата neo4j. Обработката беше со скрипта во Java и истата се искористува само еднаш во дадената задача.

Понатаму се пишуваат Cypher прашалници директно во neo4j терминал, кои всушност ќе ги внесат податоците од .csv во базата neo4j. Ова значително го забрзува импортоот, наместо преку библиотеки кои се користат за поврзување со базата.

Потребно беше да се креираат и индекси за брзо пребарување на јазлите по нивните својства.

Во моментот кога требаше да се вчитаат податоци од датотеката followers.csv, потребно беше да се ограничи бројот на редови за секоја трансакција, бидејќи датотеката беше голема 14 милиони врски, па Java heap memory се преполнуваше и крашнуваше самото вчитување. Затоа секоја трансакција ја ограничивме на 200 000 редови.

```
LOAD CSV FROM 'file:///circles.csv' AS row CREATE (c:Circle {name:row[0]})
```

```
LOAD CSV FROM 'file:///features.csv' AS row CREATE (f:Feature {name:row[0]})
```

```
LOAD CSV FROM 'file:///users.csv' AS row CREATE (u:User {name:row[0]})
```

```
CREATE INDEX FOR (c:Circle) ON c.name
```

```
CREATE INDEX FOR (u:User) ON u.name
```

```
CREATE INDEX FOR (f:Feature) ON f.name
```

```
LOAD CSV FROM 'file:///members.csv' AS row
```

```
MATCH (u:User {name: row[0]}), (c:Circle {name:row[1]}) CREATE (u)-[m:RELATION {name:'IS_MEMBER_OF'}]->(c);
```

```
:auto LOAD CSV FROM 'file:///followers.csv' AS row
```

```
    CALL{
```

```
        WITH row
```

```
        MATCH (u:User {name: row[0]}), (u1:User {name:row[1]}) CREATE (u)-[f:RELATION {name:'FOLLOWS'}]->(u1) IN TRANSACTIONS OF 200 000 ROWS;
```

```
LOAD CSV FROM 'file:///relations.csv' AS row
```

```
MATCH (u:User {name: row[0]}), (f:Feature {name:row[2]}) CREATE (u)-[r:RELATION {name:row[1]})->(f);
```

ArangoDB потребно време за импорт: 30 минути

Со програма (.NET Core, C#) се воспоставува врска со базата, со што податоците се вадат, трансформираат и вчитуваат во базата преку врската (ETL процес).

Квериња во Neo4j:

1. Колку луѓе работат во институцијата „Google“?

```
MATCH (u:User)-[:RELATION{name:"institution"}]-({name:"Google"})
RETURN COUNT(u)
```

2. Колкав процент на луѓе во институцијата „Google“ имаат работна позиција „computer“?

```
MATCH (user:User)-[:RELATION{name:"institution"}]-(:Feature{name:"Google"})
WITH user, size((user)-[:RELATION{name:"job_title"}]-(:Feature{name:"computer"})) AS title
RETURN avg(title)*100
```

3. Кој е најчест град од сите кругови?

```
MATCH (c:Circle)
CALL
{
    WITH c
    MATCH (f:Feature)-[:RELATION{name:"place"}]-()-
[:RELATION{name:"IS_MEMBER_OF"}]- (c)
    RETURN f, COUNT(f) as d
    ORDER BY COUNT(f) DESC
    LIMIT 1
}
RETURN c.name,f.name,d
ORDER BY d DESC
```

4. Кои се 10-те корисници со најмногу следбеници, и колку следбеници имаат?

```
MATCH (:User)-[:RELATION{name:"FOLLOWS"}]->(thebest:User)
RETURN thebest.name, COUNT(thebest)
ORDER BY COUNT(thebest) DESC
LIMIT 10
```

5. Колкав процент од корисниците биле на универзитет?

```

MATCH (user:User)
WITH user, size((user)-[:RELATION{name:"university"}]-()) AS relCo
RETURN avg(relCo)*100

```

6. Кој универзитет е најпопуларен кај корисниците од местото „New Jersey“?

```

MATCH (user:User)-[:RELATION{name:"place"}]-(:Feature{name:"New Jersey"})
MATCH (user)-[:RELATION{name:"university"}]-(u:Feature)
RETURN u.name, COUNT(user)
ORDER BY COUNT(user) DESC
LIMIT 1

```

Квериња во ArangoDB:

1. Колку луѓе работат во институцијата „Google“?

```

FOR institution IN Feature
FOR worksAt IN Relation
FILTER worksAt._relationName == "institution"
AND worksAt._to == institution._id
AND institution._featureName == "Google"
COLLECT WITH COUNT INTO employeeCount
RETURN employeeCount

```

2. Колкав процент на луѓе во институцијата „Google“ имаат работна позиција „computer“?

```

LET institutionEmployees = (
  FOR institution IN Feature
  FOR worksAt IN Relation
  FILTER worksAt._relationName == "institution"
  AND worksAt._to == institution._id
  AND institution._featureName == "Google"
  RETURN { id: worksAt._from }
)
FOR employee IN institutionEmployees
LET title = (
  FOR jobTitle IN Feature
  FOR hasJobTitle IN Relation
  FILTER hasJobTitle._relationName == "job_title"
  AND hasJobTitle._to == jobTitle._id
  AND jobTitle._featureName == "computer"

```

```

    AND hasJobTitle._from == employee.id
    COLLECT WITH COUNT INTO hasTitle
    RETURN { appears: hasTitle }
)
COLLECT
AGGREGATE percentage = AVG(title[0].appears)
RETURN (percentage * 100)

```

3. Кој е најчест град од сите кругови?

```

FOR circleData IN Circle
LET cityPopulation = (
  FOR cityData IN Feature
  FOR member IN IsMemberOf
  FOR lives IN Relation
  FILTER member._from == lives._from
  AND member._to == circleData._id
  AND lives._to == cityData._id
  AND lives._relationName == "place"
  COLLECT city = cityData._featureName
  WITH COUNT INTO population
  SORT population DESC
  LIMIT 1
  RETURN { "city": city, "population": population }
)
RETURN { "circle": circleData._id, "city": cityPopulation[0].city, "population":
cityPopulation[0].population }

```

4. Кои се 10-те корисници со најмногу следбеници, и колку следбеници имаат?

```

FOR user IN User
FOR followers IN Follows
FILTER followers._to == user._id
COLLECT userId = user._id WITH COUNT INTO followerCount
SORT followerCount DESC
LIMIT 10
RETURN { "userId": userId, "followers": followerCount }

```

5. Колкав процент од корисниците биле на универзитет?

```

FOR user IN User
LET uni = (
  FOR relation IN Relation
  FILTER relation._relationName == "university"
  AND relation._from == user._id

```

```

    COLLECT WITH COUNT INTO went
    RETURN { went: went }
)
COLLECT
AGGREGATE percentage = AVG(uni[0].went)
RETURN (percentage * 100)

```

6. Кој универзитет е најпопуларен кај корисниците од местото „New Jersey“?

```

FOR place IN Feature
FOR hasPlace IN Relation
FOR university IN Feature
FOR hasUniversity IN Relation
FILTER hasPlace._from == hasUniversity._from
AND hasPlace._relationName == "place"
AND hasPlace._to == place._id
AND place._featureName == "New Jersey"
AND hasUniversity._relationName == "university"
AND hasUniversity._to == university._id
COLLECT universityName = university._featureName WITH COUNT INTO
usersInUniversity
SORT usersInUniversity DESC
LIMIT 1
RETURN universityName

```

Резултати:

Neo4j Квери 1	<div> COUNT(u) </div> <div> 1572 </div> <div> Started streaming 1 records after 6 ms and completed after 119 ms. </div>
Neo4j Квери 2	<div> avg(title)*100 </div> <div> 1.145038167938932 </div> <div> Started streaming 1 records after 10 ms and completed after 486 ms. </div>

Neo4j Квери 3	<table><tr><th>c.name</th><th>l.name</th><th>d</th></tr><tr><td>"bAak0t1zU"</td><td>"San Francisco, CA"</td><td>30</td></tr><tr><td>"P88Sh6UzcY6"</td><td>"San Francisco, CA"</td><td>28</td></tr><tr><td>"h8BkcUgBaKo"</td><td>"San Francisco, CA"</td><td>21</td></tr><tr><td>"NuhYWBLgT"</td><td>"San Francisco, CA"</td><td>21</td></tr><tr><td>"875ATy6LHo"</td><td>"San Francisco, CA"</td><td>21</td></tr><tr><td>"JavWUwVC4D6"</td><td>"San Francisco, CA"</td><td>21</td></tr></table> <div>Started streaming 408 records after 57 ms and completed after 20392 ms.</div>	c.name	l.name	d	"bAak0t1zU"	"San Francisco, CA"	30	"P88Sh6UzcY6"	"San Francisco, CA"	28	"h8BkcUgBaKo"	"San Francisco, CA"	21	"NuhYWBLgT"	"San Francisco, CA"	21	"875ATy6LHo"	"San Francisco, CA"	21	"JavWUwVC4D6"	"San Francisco, CA"	21
c.name	l.name	d																				
"bAak0t1zU"	"San Francisco, CA"	30																				
"P88Sh6UzcY6"	"San Francisco, CA"	28																				
"h8BkcUgBaKo"	"San Francisco, CA"	21																				
"NuhYWBLgT"	"San Francisco, CA"	21																				
"875ATy6LHo"	"San Francisco, CA"	21																				
"JavWUwVC4D6"	"San Francisco, CA"	21																				
Neo4j Квери 4	<table><tr><th>thebest.name</th><th>COUNT(thebest)</th></tr><tr><td>"11109108952727420853"</td><td>25638</td></tr><tr><td>"104967932455782713079"</td><td>22813</td></tr><tr><td>"113455290791279442483"</td><td>19895</td></tr><tr><td>"100535338638690515335"</td><td>17246</td></tr><tr><td>"100518419853963396365"</td><td>16843</td></tr><tr><td>"106189723444098348646"</td><td>15950</td></tr></table> <div>Started streaming 10 records after 7 ms and completed after 7478 ms.</div>	thebest.name	COUNT(thebest)	"11109108952727420853"	25638	"104967932455782713079"	22813	"113455290791279442483"	19895	"100535338638690515335"	17246	"100518419853963396365"	16843	"106189723444098348646"	15950							
thebest.name	COUNT(thebest)																					
"11109108952727420853"	25638																					
"104967932455782713079"	22813																					
"113455290791279442483"	19895																					
"100535338638690515335"	17246																					
"100518419853963396365"	16843																					
"106189723444098348646"	15950																					
Neo4j Квери 5	<div>avg(relCo)*100</div> <div>8.596781412364567</div> <div>Started streaming 1 records after 7 ms and completed after 187762 ms.</div>																					
Neo4j Квери 6	<div>u.name</div> <div>"Rutgers University"</div> <div>Started streaming 1 records after 7 ms and completed after 166 ms.</div>																					

ArangoDB Квери 1	<div>Query 1 elements 55.889 ms ▼</div> <div><div>1 ▾</div><div>2</div><div>3</div></div> <div>[1572]</div>
------------------	---

ArangoDB Квери 2	<div><div>Query</div><div><div></div>1 elements</div><div><div></div>181.948 ms</div><div></div></div> <div><div>1</div><div>2</div><div>3</div></div> <div>[1.1450381679389312]</div>																						
ArangoDB Квери 3	<div><table><tr><th>ID</th><th>Query String</th><th>Bind parameters</th><th>User</th><th>Runtime</th></tr><tr><td>40431131</td><td><pre>FOR circleData IN circle LET cityPopulation = (FOR cityData IN feature FOR member IN lookupOf FOR lives IN relation FILTER member._from == lives._from AND member._to == circleData._id AND lives._to == cityData._id AND lives._relationName == "place" COLLECT city = cityData._featureName WITH COUNT INTO population SORT population DESC LIMIT 1 RETURN ("city": city, "population": population)) RETURN ("circle": circleData._id, "city": cityPopulation[0].city, "population": cityPopulation[0].population)</pre></td><td>()</td><td>root</td><td>1005.23s</td></tr></table><p>После 1000 секунди, кверито сеуште не беше завршено и беше стопирано – предолго трае.</p></div>	ID	Query String	Bind parameters	User	Runtime	40431131	<pre>FOR circleData IN circle LET cityPopulation = (FOR cityData IN feature FOR member IN lookupOf FOR lives IN relation FILTER member._from == lives._from AND member._to == circleData._id AND lives._to == cityData._id AND lives._relationName == "place" COLLECT city = cityData._featureName WITH COUNT INTO population SORT population DESC LIMIT 1 RETURN ("city": city, "population": population)) RETURN ("circle": circleData._id, "city": cityPopulation[0].city, "population": cityPopulation[0].population)</pre>	()	root	1005.23s												
ID	Query String	Bind parameters	User	Runtime																			
40431131	<pre>FOR circleData IN circle LET cityPopulation = (FOR cityData IN feature FOR member IN lookupOf FOR lives IN relation FILTER member._from == lives._from AND member._to == circleData._id AND lives._to == cityData._id AND lives._relationName == "place" COLLECT city = cityData._featureName WITH COUNT INTO population SORT population DESC LIMIT 1 RETURN ("city": city, "population": population)) RETURN ("circle": circleData._id, "city": cityPopulation[0].city, "population": cityPopulation[0].population)</pre>	()	root	1005.23s																			
ArangoDB Квери 4	<div><table><tr><th>userid</th><th>followers</th></tr><tr><td>User/11109108952727420853</td><td>25638</td></tr><tr><td>User/104987932455782713675</td><td>22813</td></tr><tr><td>User/113455290791279442483</td><td>19895</td></tr><tr><td>User/100535338638690515335</td><td>17246</td></tr><tr><td>User/100518419853963396365</td><td>16843</td></tr><tr><td>User/106189723444098348646</td><td>15950</td></tr><tr><td>User/100962871525684315897</td><td>15798</td></tr><tr><td>User/115360471097759949621</td><td>15744</td></tr><tr><td>User/105237212888595777019</td><td>15594</td></tr><tr><td>User/116247667398036716276</td><td>15459</td></tr></table><div><div>Query</div><div><div></div>10 elements</div><div><div></div>14.426 s</div><div></div></div></div>	userid	followers	User/11109108952727420853	25638	User/104987932455782713675	22813	User/113455290791279442483	19895	User/100535338638690515335	17246	User/100518419853963396365	16843	User/106189723444098348646	15950	User/100962871525684315897	15798	User/115360471097759949621	15744	User/105237212888595777019	15594	User/116247667398036716276	15459
userid	followers																						
User/11109108952727420853	25638																						
User/104987932455782713675	22813																						
User/113455290791279442483	19895																						
User/100535338638690515335	17246																						
User/100518419853963396365	16843																						
User/106189723444098348646	15950																						
User/100962871525684315897	15798																						
User/115360471097759949621	15744																						
User/105237212888595777019	15594																						
User/116247667398036716276	15459																						
ArangoDB Квери 5	<div><div>Query</div><div><div></div>1 elements</div><div><div></div>3.711 s</div><div></div></div> <div><div>1</div><div>2</div><div>3</div></div> <div>[8.596781412364514]</div>																						
ArangoDB Квери 6	<div><div>Query</div><div><div></div>1 elements</div><div><div></div>19.904 s</div><div></div></div> <div><div>1</div><div>2</div><div>3</div></div> <div>["Rutgers University"]</div>																						