Tags

Users

adges

Inanswered

Ask Question

Ten. Million. Questions. Let's celebrate all we've done together.

Stack Overflow is a question and answer site for professional and enthusiast programmers. It's 100% free.

Take the 2-minute tour

asked 4 months ago

active 4 months ago

viewed 74 times

## find paths with specific length combined with nodes with specific degree neo4j



Blog

Why Stack Overflow is a Good Workplace for Women

0

I am trying to write a query to find all nodes with outdegree X and only return the paths that contains those nodes when path length is equal to Y

If I want to get only nodes with outdegree X I use the following Cypher query

MATCH (s:URL)-[r:VISITED\*]->(t:URL)
WITH s, count(t) as degreeout
WHERE 73 in s.job\_id and degreeout <4
return s, degreeout

If I want to get only paths with length = X I use the following query

MATCH p=(s:URL)-[r:VISITED\*]->(t:URL)
WHERE length(p)=7
return p

I tried the combine the previous two queries in the following query

MATCH (s:URL)-[r:VISITED\*]->(t:URL) WITH s, COLLECT(DISTINCT id(s)) as matched, count(t) as degreeout WHERE 73 in s.job\_id and degreeout <4 MATCH p=(s2:URL)-[r:VISITED\*]-(t2:URL) WHERE id(s2) in matched and length(p) >=1 RETURN p

Whenever I execute the query, the machine keeps processing and then I get an error no enough memory.

It seems like there is an infinite loop!!

neo4j cypher graph-databases
share improve this question

asked Apr 13 at 13:01 I.el-sayed 65 • 1 • 10

active

oldest

votes

add a comment

## 1 Answer

If you're just interested in traversing the relationship an exact number of times you can include that in the path expression:



MATCH p=(s:URL)-[r:VISITED\*7]->(t:URL)
return p



In general you should avoid doing traversals of unlimited length, i.e. :VISITED\*. If you want to keep the depth variable because it's unknown it's good practice to set a max. value, i.e. :VISITED\*..7.

If I understood correctly, your original query can be adjusted, just be setting the variable length to 7 in the path:

MATCH (s:URL)-[r:VISITED\*7]->(t:URL)
WITH s, count(t) as degreeout
WHERE 73 in s.job\_id and degreeout <4
return s, degreeout

#### Related

- 1 Optimize Neo4j Cypher path finding with limited paths in an undirected graph
- cypher query to find paths between set of nodes takes a very long time
- 2 Find all paths with smallest weight if there are many relations between two nodes using cypher in Neo4j
- Extract all nodes from a path query with cypher (from Neo4j)?
- Neo4j first node to meet relationship in a movie model
- o Finding Paths with Neo4j Cypher Below a Given Length, but Excluding Those with an Nodes with a Specific Property set to a Specific Value
- 2 Neo4j: Sum relationship properties where node properties equal Value A and Value B (intersection)
- Find the path between "n" nodes in neo4J using cypher or java
- 1 Cypher neo4j find recursive end nodes

You should see some performance improvement because now paths of length > 7 will be excluded from the results and they won't be traversed. Again, always avoid unlimited depth traversals unless there is a really good reason for it and you have enough computer resources + time for the query to complete.

Regarding performance best practices, this query will still not perform very well, since it's forcing a graph scan to find the start node. I understand the nodes labeled URL contain a property <code>job\_id</code> of type Array because of the <code>in</code> operator? Neo4j needs to read all URL nodes and their properties, then scan through those arrays just to find the start node.

I would recommend changing your data model to use schema indexes based on an exact property value. Example:

(j:Job {job\_id: 73})-[:SOMETHING?]->(u:URL {...})

We would also add a schema index:

CREATE INDEX ON :Job(job\_id)

Then you can query like this:

MATCH (j:Job {job\_id: 73})-[:SOMETHING?]->(s:URL)-[r:VISITED\*7]->(t:URL)
WITH s, count(t) as degreeout
WHERE degreeout <4
return s, degreeout

share improve this answer

answered Apr 13 at 14:38

albertoperdomo
1.132 • 3 • 9

I label the starting node as STARTING\_URL ... so the starting node will have two labels "URL" and "STARTING\_URL" – I.el-sayed Apr 14 at 6:19  $\mathscr{I}$ 

Regarding the job\_id I can't have it as an int property value. I should have it as an array property. I don't think that this way (j:Job {job\_id: 73}) (property as dictionary) would enable me to handle the job\_id attribute as an array ... !?? - I.el-sayed | Apr 14 at 6:29 | P

The problem with that approach is that locating a node by an array property is not efficient, since it won't use any indexes and Neo4j will have to scan all nodes with matching labels. If you have n job\_ids for each STARTING\_URL, then you can probably create one node for each job\_id and have a relationship to the STARTING\_URL as I tried to explain in the last section of my answer. This approach would take more advantage of Neo4j's graph performance capabilities. – albertoperdomo Apr 14 at 18:50 \$\mathbb{N}\$

I did create the JOB node as u suggested and changed the design. I just have a question now. How I can return all paths with length >=3 and the same time exclude all nodes with outdegree > 10?? How I can do this in a cypher query...?? – I.el-sayed | Apr 14 at 23:16

You can add a minimum depth: MATCH (j:Job {job\_id: 73})-[:SOMETHING?]->(s:URL)-[r:VISITED\*3..]->(t:URL) WITH s, count(t) as degreeout WHERE degreeout <11 return s, degreeout — albertoperdomo Apr 15 at 9:01

show 1 more comment

### Your Answer



#### Sign up or log in



## Post as a guest

Name	
Email	
required, but never shown	

O Cypher/Neo4j: How to match nodes related to n "consecutive" nodes

#### Hot Network Questions

- Name for a device purposefully put together from faulty parts
- I added more water to my ciabatta dough by accident
- For loop formatting
- Does touching the tent outer wall from the inside make it leak?
- What are some techniques to avoid touching a wet tent wall?
- If static methods can't be overriden, how its working here (For Java)?
- Approximation of Borel sets by a countable collection of majorants
- Why did Holly decide to simply wait out the radioactivity?
- What is aliasing and what causes it?
- Could a communication satellite that fails to reach GEO be repurposed?
- A short, brutal riddle
- Sign that word!
- What's the philosophy behind delaying writing data to disk
- Using Emergency Fund to Sell Upside-down Car
- Lunch with the CEO
- Does developing on ARM
  Processors from different vendors
  differ much?
- The Hiding Rules Contradict
  Themselves. What Can I Make Of
  This?
- Remove a language from the "Preferred language order"
- Select polygons one by one and export them using arcpy
- Combining several symmetric ciphers using XOR
- Preventing a Print expression from generating a new cell each time it executed
- Draw horizontal lines like open and close brackets
- Where did the nickname of 'Bones' for McCoy come from?
- Which type of drill bit for which iobs?

# Post Your Answer

By posting your answer, you agree to the privacy policy and terms of service.

Not the answer you're looking for? Browse other questions tagged neo4j cypher graph-databases or ask your own question.

a question feed

ECHNOLOGY			LIFE / ARTS	CULTURE / RECREATION	SCIENCE	OTHER
tack Overflow	Programmers		English Language &	Mathematics	Stack Apps	
erver Fault	Unix & Linux	Administrators	Science Fiction &	Usage Skeptics	Cross Validated (stats) cs Theoretical Computer	Meta Stack Exchange
Super User	Ask Different (Apple)	Drupal Answers	Fantasy			Area 51
Veb Applications	WordPress Development	SharePoint	Graphic Design	Mi Yodeya (Judaism)		Stack Overflow Careers
sk Ubuntu		User Experience	Movies & TV	Travel	Physics	
/ebmasters	Geographic Information	Mathematica	Seasoned Advice	Christianity	MathOverflow	
	Systems	Salesforce	Home Improvement  Personal Finance &	Arqade (gaming)	Chemistry	
Same Development	Electrical Engineering	ExpressionEngine®		Bicycles	Biology	
TeX - LaTeX	Android Enthusiasts	Answers		Role-playing Games	more (5)	
	Information Security	more (13)	Money	more (21)		
			Academia	111016 (21)		
			more (9)			