

BI / read / 18

BI 1
BI 2
BI 3
BI 4
BI 5
BI 6
BI 7
BI 8
BI 9
BI 10
BI 11
BI 12
BI 13
BI 14
BI 15
BI 16
BI 17
BI 18
BI 19
BI 20

query	BI / read / 18													
title	Friend recommendation													
pattern	<div><div>For each person1 compute top-k(person2) based on mutualFriendCount</div><pre>graph LR person1[person1: Person] -- knows --> Person[Person] Person -- knows --> person2[person2: Person] person2 -- hasInterest --> tag[tag: Tag] tag -- "name = \$tag" --> tag person1 -. "«neg» knows" .-> person2</pre></div>													
desc.	<p>For a given Person (person1) and a Tag (tag), recommend new friends (person2) who</p> <ul style="list-style-type: none">• do not yet know person1• have many mutual friends with person1• are interested in tag. <p>Rank Persons person2 based on the number of mutual friends.</p>													
params	<table><tr><td>1</td><td>person1Id</td><td>ID</td><td>Persons with a similar amount of friends are selected</td></tr><tr><td>2</td><td>tag</td><td>Long String</td><td>Tags with a similar amount of Messages are selected</td></tr></table>	1	person1Id	ID	Persons with a similar amount of friends are selected	2	tag	Long String	Tags with a similar amount of Messages are selected					
1	person1Id	ID	Persons with a similar amount of friends are selected											
2	tag	Long String	Tags with a similar amount of Messages are selected											
result	<table><tr><td>1</td><td>person2.id</td><td>ID</td><td>R</td><td></td></tr><tr><td>2</td><td>mutualFriendCount</td><td>32-bit Integer</td><td>A</td><td></td></tr></table>	1	person2.id	ID	R		2	mutualFriendCount	32-bit Integer	A				
1	person2.id	ID	R											
2	mutualFriendCount	32-bit Integer	A											
sort	<table><tr><td>1</td><td>mutualFriendCount</td><td>↓</td><td></td></tr><tr><td>2</td><td>person2.id</td><td>↑</td><td></td></tr></table>	1	mutualFriendCount	↓		2	person2.id	↑						
1	mutualFriendCount	↓												
2	person2.id	↑												
limit	20													
CPs	2.5, 8.1													