

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 --&gt; Person[Person]     Person -- knows --&gt; person2[person2: Person]     person1 -. «neg» knows .-&gt; person2     Person -- hasInterest --&gt; tag[tag: Tag]     tag -- name = \$tag</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"><li>• do not yet know person1</li><li>• have many mutual friends with person1</li><li>• are interested in tag.</li></ul> <p>Rank Persons person2 based on the number of mutual friends.</p>                       |                |   |  |  |
| params  | <div><div>1</div>person1Id</div>   | ID             | Persons with a similar amount of friends are selected |  |  |
|         | <div><div>2</div>tag</div>   | Long String    | Tags with a similar amount of Messages are selected   |  |  |
| result  | <div><div>1</div>person2.id</div>  | ID             | R   |  |  |
|         | <div><div>2</div>mutualFriendCount</div>   | 32-bit Integer | A   |  |  |
| sort    | <div><div>1</div>mutualFriendCount</div>   | ↓              |   |  |  |
|         | <div><div>2</div>person2.id</div>  | ↑              |   |  |  |
| limit   | 20   |                |   |  |  |
| CPs     | 2.5, 8.1   |                |   |  |  |