

Interactive / complex / 14

IC 1	query	Interactive / complex / 14			
IC 2	title	Trusted connection paths			
IC 3	pattern	<div><div><div>Enumerate all unweighted shortest paths on knows edges from person1 to person2.</div><div><div>person1: Person</div><div>knows*</div><div>person2: Person</div></div><div><div>id = \$person1Id</div><div>id = \$person2Id</div></div></div><div><div>Case 1: Replies on Posts, weight += 1.0 × count(c)</div><div><div>personA: Person</div><div>knows</div><div>personB: Person</div></div><div><div>hasCreator</div><div>↑</div><div>c: Comment</div><div>replyOf</div><div>post: Post</div></div></div><div><div>Case 2: Replies on Comments, weight += 0.5 × count(c1)</div><div><div>personA: Person</div><div>knows</div><div>personB: Person</div></div><div><div>hasCreator</div><div>↑</div><div>c1: Comment</div><div>replyOf</div><div>c2: Comment</div></div></div><div><div>For each edge on the path, calculate a weight based on interactions between the pair of Persons of the edge, are calculated as a sum of cases #1 and #2 for the Persons (both ways), and the sum of these weights determine the total weight of each path.</div><div><div>p1</div><div>knows</div><div>pX</div><div>knows</div><div>pY</div><div>...</div><div>pW</div><div>knows</div><div>p2</div></div></div></div>			
IC 4					
IC 5					
IC 6					
IC 7					
IC 8					
IC 9					
IC 10					
IC 11					
IC 12					
IC 13					
IC 14					
	desc.	<p>Given two Persons, find all (unweighted) shortest paths between these two Persons, in the subgraph induced by the knows relationship.</p> <p>Then, for each path calculate a weight. The nodes in the path are Persons, and the weight of a path is the sum of weights between every pair of consecutive Person nodes in the path.</p> <p>The weight for a pair of Persons is calculated based on their interactions:</p> <ul style="list-style-type: none">• Every direct reply (by one of the Persons) to a Post (by the other Person) contributes 1.0.• Every direct reply (by one of the Persons) to a Comment (by the other Person) contributes 0.5. <p>Note that interactions are counted both ways (e.g. if Alice writes 2 Post replies and 1 Comment reply to Bob, while Bob writes 3 Post replies and 4 Comment replies to Alice, their interaction score is $2 \times 1.0 + 1 \times 0.5 + 3 \times 1.0 + 4 \times 0.5 = 7.5$).</p> <p>Return all the paths with shortest length, and their weights. Do not return any rows if there is no path between the two Persons.</p>			
	params	<div><div>1</div><div>person1Id</div><div>ID</div></div> <div><div>2</div><div>person2Id</div><div>ID</div></div>			
	result	<div><div>1</div><div>personIdsInPath</div><div>[ID]</div></div> <div><div>2</div><div>pathWeight</div><div>64-bit Float</div></div>	<div>C</div> <div>C</div>	<div>identifiers representing an ordered sequence of the Persons in the path</div>	
	sort	<div><div>1</div><div>pathWeight</div><div>↓</div></div>	<div>The order of paths with the same weight is unspecified</div>		
	CPs	<div>3.3, 5.3, 7.2, 7.3, 7.5, 7.7, 8.1, 8.2, 8.3, 8.6</div>			
	relevance	<div>This query looks for a variable length path, starting at a given Person and finishing at an another given Person. This is a more complex query as it not only requires computing the path length, but returning it and computing a weight. To compute this weight one must look for smaller sub-queries with paths of length three, formed by the two Persons at each step, a Post and a Comment.</div>			