BI / read / 16

BI 1	query	BI / read / 16
BI 2	title	Fake news detection
BI 3		For \$tagX/\$dayX in [tagA/dateA, tagB/dateB], compute scoreX = count(messageX)
BI 5		Create an induced subgraph of Persons who created a Message with Tag \$tagX on \$dateX
BI 6		tag: Tag Message person: Person
BI 7		name = \$tagX hasTag day(creationDate) = \$dateX
BI 8		O In the output the Manager (voice the same and this of free Dearle with a Constitute of the same
BI 9	pattern	2. In the subgraph, count the Messages (using the same conditions) from People with ≤ \$maxKnowsLimit friends count(messageX)
BI 10		tag: Tag messageX: Message hasCreator
BI 11		name = \$tagX day(creationDate) = \$dateX count ≤ \$maxKnowsLimit
BI 12		«opt» knows
BI 13		Person
BI 15		
BI 16		Given two Tag/date pairs (tagA/dateA and tagB/dateB), for each pair tagX/dateX:
BI 17		Create an induced subgraph between Persons where for each pair of Persons person1/person2,
BI 18		both have created a Message on the day of dateX with Tag tagX.
BI 19		• In the induced subgraph, only keep pairs of Persons who have at most maxKnowsLimit friends
BI 20	(in the induced subgraph).For these Persons, count the number of Messa	
		• For these Persons, count the number of Messages created on dateX with Tag tagX.
		Poturn Down who had at least one Manager for both to add to and to and to an army ranked by their
		Return Persons who had at least one Messages for both tagA/dateA and tagB/dateB ranked by their total number of Messages (descending)
		1 tagA Long String
		2 dateA Date
	params	3 tagB Long String
		4 dateB Date
		5 maxKnowsLimit 32-bit Integer Selected between 3 and 6
		1 person.id ID R
	result	2 messageCountA 32-bit Integer A Message count for tagA/dateA
		3 messageCountB 32-bit Integer A Message count for tagB/dateB
		171CSSage Count for Cagurdaces
		messageCountA +
	sort	messageCountB
		2 person.id ↑
	limit	20
	CPs	5.3, 8.4, 8.5
		There are two major ways to compute this query: (1) create the induced subgraph as suggested by the specification
	relevance	(either as a view or in materialized form), or (2) skip creating the induced subgraph and perform on-the-fly check
		for the number of friends (who also posted at least one Message with the given Tag on the given date). The latter
		approach is easier to express in systems which do not provide graph views but might result in redundant computations (the query engine will might repeatedly check whether a Person has at least one Message that satisfies the conditions).
		V - 1 - 7 - 8