TRAJECTORY MATCHING

L’operatore di trajectory matching ha lo scopo di calcolare una misura di similarità tra oggetti JSON di natura eterogenea contenenti una sequenza di coordinate spaziali (traccia). La sintassi è la seguente:

*TRAJECTORY MATCHING collectionReference1, collectionReference2*

*(*

*PARTITION selectionCondition*

*(*

*MATCHING fieldReference WRT fieldReference THRESHOLD (unit) value*

*(WHERE selectionCondition)\**

*INTO fieldName*

*(MIN\_SIMILARITY value)\**

*)+*

*)+*

*(KEEP OTHERS | DROP OTHERS );*

Questo operatore ha una struttura simile a quella del GROUP ma lavora su due collezioni distinte definite in seguito per chiarezza *target* e *input*. Questo aspetto definisce la natura asimmetrica dell’operatore.

In particolare la prima collezione specificata (*collectionReference1*) rappresenta il target e la seconda (*collectionReference2*) rappresenta l’input.

Prima di tutto il PARTITION ricerca nella collezione *target* gli elementi che soddisfano la *selectionCondition* i quali saranno usati per il partizionamento.

Successivamente nel MATCHING vengono specificati il campo del targete WRT il campo dell’*input* che contengono le sequenze di coordinate spaziali tra cui verrà calcolata la similarità.

La soglia di match THRESHOLD definisce inoltre quanto due punti debbano essere vicini affinché siano considerati simili.

Nel WHERE è possibile eventualmente selezionare solo alcuni oggetti dell’*input*.

A questo punto sono stati definiti tutti gli elementi dell’operazione e l’algoritmo provvederà per ogni coppia di tracce del *target* e dell’*input* a calcolare la loro misura di similarità. Il risultato di questa operazione verrà raggruppato in un nuovo campo specificato nell’INTO.

È anche possibile mantenere nella collezione in output solo alcuni match che hanno ottenuto una misura di similarità >= di MIN\_SIMILARITY.

La capacità di questo operatore di lavorare con oggetti eterogenei (coerentemente con gli altri operatori di J-CO) è data sia dalla possibilità di specificare più condizioni di MATCHING per la stessa partizione che di specificare più PARTITION.

Nel caso in cui un oggetto della collezione *input* trovi una corrispondenza in più PARTITION verrà inserito solo nella prima che lo contiene.

Infine con (KEEP OTHERS | DROP OTHERS) è possibile specificare se mantenere nella collezione temporanea in output gli oggetti i quali non hanno una corrispondenza in nessuna PARTITION.

Di seguito viene riportato un esempio pratico di applicazione dell’operatore.

*studenti*

*[*

*{*

*“name”: ”studente1”, “traceWalk” : [],*

*“traceTrain”:[]*

*},*

*{*

*“name”:”studente2”, “traceWalk” : [ ],*

*“traceTrain”:[]*

*},*

*...*

*]*

*percorsi*

*[*

*{*

*“walk”: “StazioneBG-Salvecchio”,*

*“trace”: []*

*},*

*{*

*“walk”: “StazioneBG-Caniana”,*

*“trace”: []*

*},*

*{*

*“trasportoPubblico”: “Milano-Bergamo”,*

*“stops”: []*

*},*

*{*

*“trasportoPubblico”: “Brescia-Bergamo”*

*“stops”: []*

*}*

*]*

TRAJECTORY MATCHING [studenti@MyDB](mailto:studenti@MyDB), [percorsi@MyDB](mailto:percorsi@MyDB)

PARTITION WITH studenti.traceWalk, studenti.traceTrain

MATCHING studenti.traceWalk WRT percorsi.trace THRESHOLD (M) 100

WHERE WITH percorsi.walk

INTO matchingWalk MIN\_SIMILARITY 0,4

MATCHING studenti.traceTrain WRT percorsi.stops THRESHOLD (M) 10

WHERE WITH percorsi.trasportoPubblico

INTO matchingTrain MIN\_SIMILARITY 0,4

DROP OTHERS;

Viene creata una partizione di oggetti del *target* studenticontenente gli elementi che soddisfano la selectionCondition (*WITH studenti.traceWalk, studenti.traceTrain)*.

Successivamente si specifica quali sono gli argomenti del MATCHING, in questo caso voglio trovare la similarità tra:

- il campo traceWalk (*target*) rispetto al campo trace (*input*) degli oggetti dell’input che contengono il campo “walk”

-il campo studenti.traceTrain (*target*) rispetto al campo stops (*input*) degli oggetti dell’*input* che contengono il campo “trasportoPubblico”

per ognuno dei quali è specificato il THRESHOLD e il campo INTO dentro cui gli oggetti dell’*input* verranno raggruppati.

La collezione temporanea prodotto in output avrà la seguente struttura:

*[*

*{*

*"name": "studente2",*

*"traceWalk": [],*

*"traceTrain":[],*

*"matchingWalk": [*

*{*

*"walk":"StazioneBG-Salvecchio",*

*"trace":[]*

*"similarity":0,5*

*}],*

*"matchingTrain": [*

*{*

*"trasportoPubblico":"Brescia-Bergamo",*

*"stops": []*

*"similarity" : 0,7*

*}]*

*}*

*]*

Questo risultato significa che:

- il campo traceWalk di studente2 ha una similarità del 50% con la traccia del percorso “StazioneBG-Salvecchio”

-il campo traceTrain di studente2 ha una similarità del 70% con le fermate della linea “Brescia-Bergamo”.

Gli oggetti in cui entrambi i campi “matchingWalk” e “matchingTrain” sono vuoti (per i quali la similarità è inferiore al 40%) non vengono riportati nella collezione temporanea in output.

Un esempio di come apparirebbe il documento se MIN\_SIMILARITY non fosse specificato è riportato di seguito.

*[*

*{*

*"name": "studente1",*

*"traceWalk": [],*

*"traceTrain":[],*

*"matchingWalk": [*

*{*

*"walk":"StazioneBG-Salvecchio",*

*"trace":[]*

*"similarity":0,2*

*},*

*{*

*"walk":"StazioneBG-Caniana",*

*"trace":[]*

*"similarity":0,1*

*}],*

*"matchingTrain": [*

*{*

*"trasportoPubblico":"Brescia-Bergamo",*

*"stops": []*

*"similarity" : 0,3*

*},*

*{*

*"trasportoPubblico":"Milano-Bergamo",*

*"stops": []*

*"similarity" : 0,1*

*}]*

*},*

*{*

*"name": "studente2",*

*"traceWalk": [],*

*"traceTrain":[],*

*"matchingWalk": [*

*{*

*"walk":"StazioneBG-Salvecchio",*

*"trace":[]*

*"similarity":0,5*

*},*

*{*

*"walk":"StazioneBG-Caniana",*

*"trace":[]*

*"similarity":0,1*

*}],*

*"matchingTrain": [*

*{*

*"trasportoPubblico":"Brescia-Bergamo",*

*"stops": []*

*"similarity" : 0,7*

*},*

*{*

*"trasportoPubblico":"Milano-Bergamo",*

*"stops": []*

*"similarity" : 0,2*

*}]*

*},*

*…*

*]*