package com.twitter.simclusters\_v2.common.clustering

import com.twitter.simclusters\_v2.common.UserId

import com.twitter.simclusters\_v2.thriftscala.NeighborWithWeights

class MaxFavScoreRepresentativeSelectionMethod[T] extends ClusterRepresentativeSelectionMethod[T] {

/\*\*

\* Identify the member with largest favScoreHalfLife100Days and return it.

\*

\* @param cluster A set of NeighborWithWeights.

\* @param embeddings A map of producer ID -> embedding.

\*/

def selectClusterRepresentative(

cluster: Set[NeighborWithWeights],

embeddings: Map[UserId, T],

): UserId = {

val key = cluster.maxBy { x: NeighborWithWeights => x.favScoreHalfLife100Days.getOrElse(0.0) }

key.neighborId

}

}