package com.twitter.follow\_recommendations.common.candidate\_sources.salsa

import com.twitter.follow\_recommendations.common.clients.real\_time\_real\_graph.RealTimeRealGraphClient

import com.twitter.hermit.model.Algorithm

import com.twitter.product\_mixer.core.model.common.identifier.CandidateSourceIdentifier

import com.twitter.stitch.Stitch

import javax.inject.Inject

import javax.inject.Singleton

@Singleton

class RecentEngagementDirectFollowSalsaExpansionSource @Inject() (

realTimeRealGraphClient: RealTimeRealGraphClient,

salsaExpander: SalsaExpander)

extends SalsaExpansionBasedCandidateSource[Long](salsaExpander) {

override val identifier: CandidateSourceIdentifier =

RecentEngagementDirectFollowSalsaExpansionSource.Identifier

override def firstDegreeNodes(target: Long): Stitch[Seq[Long]] = realTimeRealGraphClient

.getUsersRecentlyEngagedWith(

target,

RealTimeRealGraphClient.EngagementScoreMap,

includeDirectFollowCandidates = true,

includeNonDirectFollowCandidates = false

).map { recentlyFollowed =>

recentlyFollowed

.take(RecentEngagementDirectFollowSalsaExpansionSource.NumFirstDegreeNodesToRetrieve)

.map(\_.id)

}

override def maxResults(target: Long): Int =

RecentEngagementDirectFollowSalsaExpansionSource.OutputSize

}

object RecentEngagementDirectFollowSalsaExpansionSource {

val Identifier: CandidateSourceIdentifier = CandidateSourceIdentifier(

Algorithm.RecentEngagementSarusOcCur.toString)

val NumFirstDegreeNodesToRetrieve = 10

val OutputSize = 200

}