package com.twitter.timelines.prediction.features.two\_hop\_features

import com.twitter.dal.personal\_data.thriftjava.PersonalDataType

import com.twitter.graph\_feature\_service.thriftscala.{EdgeType, FeatureType}

object TwoHopFeaturesConfig {

val leftEdgeTypes = Seq(EdgeType.Following, EdgeType.Favorite, EdgeType.MutualFollow)

val rightEdgeTypes = Seq(

EdgeType.FollowedBy,

EdgeType.FavoritedBy,

EdgeType.RetweetedBy,

EdgeType.MentionedBy,

EdgeType.MutualFollow)

val edgeTypePairs: Seq[(EdgeType, EdgeType)] = {

for (leftEdgeType <- leftEdgeTypes; rightEdgeType <- rightEdgeTypes)

yield (leftEdgeType, rightEdgeType)

}

val featureTypes: Seq[FeatureType] = edgeTypePairs.map(pair => FeatureType(pair.\_1, pair.\_2))

val personalDataTypesMap: Map[EdgeType, Set[PersonalDataType]] = Map(

EdgeType.Following -> Set(PersonalDataType.CountOfFollowersAndFollowees),

EdgeType.Favorite -> Set(

PersonalDataType.CountOfPrivateLikes,

PersonalDataType.CountOfPublicLikes),

EdgeType.MutualFollow -> Set(PersonalDataType.CountOfFollowersAndFollowees),

EdgeType.FollowedBy -> Set(PersonalDataType.CountOfFollowersAndFollowees)

)

}