package com.twitter.timelineranker.common

import com.twitter.servo.util.FutureArrow

import com.twitter.timelineranker.core.HydratedCandidatesAndFeaturesEnvelope

import com.twitter.timelines.earlybird.common.utils.EarlybirdFeaturesHydrator

import com.twitter.util.Future

object InNetworkTweetsSearchFeaturesHydrationTransform

extends FutureArrow[

HydratedCandidatesAndFeaturesEnvelope,

HydratedCandidatesAndFeaturesEnvelope

] {

override def apply(

request: HydratedCandidatesAndFeaturesEnvelope

): Future[HydratedCandidatesAndFeaturesEnvelope] = {

Future

.join(

request.candidateEnvelope.followGraphData.followedUserIdsFuture,

request.candidateEnvelope.followGraphData.mutuallyFollowingUserIdsFuture

).map {

case (followedIds, mutuallyFollowingIds) =>

val featuresByTweetId = EarlybirdFeaturesHydrator.hydrate(

searcherUserId = request.candidateEnvelope.query.userId,

searcherProfileInfo = request.userProfileInfo,

followedUserIds = followedIds,

mutuallyFollowingUserIds = mutuallyFollowingIds,

userLanguages = request.userLanguages,

uiLanguageCode = request.candidateEnvelope.query.deviceContext.flatMap(\_.languageCode),

searchResults = request.candidateEnvelope.searchResults,

sourceTweetSearchResults = request.candidateEnvelope.sourceSearchResults,

tweets = request.candidateEnvelope.hydratedTweets.outerTweets,

sourceTweets = request.candidateEnvelope.sourceHydratedTweets.outerTweets

)

request.copy(features = featuresByTweetId)

}

}

}