package com.twitter.home\_mixer.product.scored\_tweets.response\_transformer

import com.twitter.home\_mixer.model.HomeFeatures.AuthorIdFeature

import com.twitter.home\_mixer.model.HomeFeatures.CandidateSourceIdFeature

import com.twitter.home\_mixer.model.HomeFeatures.FromInNetworkSourceFeature

import com.twitter.home\_mixer.model.HomeFeatures.IsRetweetFeature

import com.twitter.home\_mixer.model.HomeFeatures.SourceTweetIdFeature

import com.twitter.home\_mixer.model.HomeFeatures.SourceUserIdFeature

import com.twitter.home\_mixer.model.HomeFeatures.SuggestTypeFeature

import com.twitter.product\_mixer.core.feature.Feature

import com.twitter.product\_mixer.core.feature.featuremap.FeatureMap

import com.twitter.product\_mixer.core.feature.featuremap.FeatureMapBuilder

import com.twitter.product\_mixer.core.functional\_component.transformer.CandidateFeatureTransformer

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

import com.twitter.timelineservice.{thriftscala => t}

import com.twitter.timelineservice.suggests.{thriftscala => st}

import com.twitter.timelineservice.suggests.logging.candidate\_tweet\_source\_id.{thriftscala => cts}

object ScoredTweetsListsResponseFeatureTransformer extends CandidateFeatureTransformer[t.Tweet] {

override val identifier: TransformerIdentifier =

TransformerIdentifier("ScoredTweetsListsResponse")

override val features: Set[Feature[\_, \_]] = Set(

AuthorIdFeature,

CandidateSourceIdFeature,

FromInNetworkSourceFeature,

IsRetweetFeature,

SuggestTypeFeature,

SourceTweetIdFeature,

SourceUserIdFeature,

)

override def transform(candidate: t.Tweet): FeatureMap = {

FeatureMapBuilder()

.add(AuthorIdFeature, candidate.userId)

.add(CandidateSourceIdFeature, Some(cts.CandidateTweetSourceId.ListTweet))

.add(FromInNetworkSourceFeature, false)

.add(IsRetweetFeature, candidate.sourceStatusId.isDefined)

.add(SuggestTypeFeature, Some(st.SuggestType.RankedListTweet))

.add(SourceTweetIdFeature, candidate.sourceStatusId)

.add(SourceUserIdFeature, candidate.sourceUserId)

.build()

}

}