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

import com.twitter.explore\_ranker.{thriftscala => ert}

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.HasVideoFeature

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

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

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.suggests.logging.candidate\_tweet\_source\_id.{thriftscala => cts}

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

object ScoredTweetsPopularVideosResponseFeatureTransformer

extends CandidateFeatureTransformer[ert.ExploreTweetRecommendation] {

override val identifier: TransformerIdentifier =

TransformerIdentifier("ScoredTweetsPopularVideosResponse")

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

AuthorIdFeature,

CandidateSourceIdFeature,

FromInNetworkSourceFeature,

HasVideoFeature,

IsRandomTweetFeature,

StreamToKafkaFeature,

SuggestTypeFeature

)

override def transform(candidate: ert.ExploreTweetRecommendation): FeatureMap = {

FeatureMapBuilder()

.add(AuthorIdFeature, candidate.authorId)

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

.add(FromInNetworkSourceFeature, false)

.add(HasVideoFeature, candidate.mediaType.contains(ert.MediaType.Video))

.add(IsRandomTweetFeature, false)

.add(StreamToKafkaFeature, true)

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

.build()

}

}