package com.twitter.home\_mixer.product.for\_you.response\_transformer

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

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

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.{thriftscala => st}

import com.twitter.search.earlybird.{thriftscala => eb}

object TweetPreviewResponseFeatureTransformer

extends CandidateFeatureTransformer[eb.ThriftSearchResult] {

override val identifier: TransformerIdentifier =

TransformerIdentifier("TweetPreviewResponse")

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

Set(AuthorIdFeature, IsTweetPreviewFeature, SuggestTypeFeature)

def transform(

input: eb.ThriftSearchResult

): FeatureMap = {

FeatureMapBuilder()

.add(IsTweetPreviewFeature, true)

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

.add(AuthorIdFeature, input.metadata.map(\_.fromUserId))

.build()

}

}