package com.twitter.product\_mixer.component\_library.candidate\_source.tweetconvosvc

import com.twitter.product\_mixer.component\_library.model.candidate.TweetCandidate

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.SuggestType

object AuthorIdFeature extends Feature[TweetCandidate, Option[Long]]

object AncestorIdsFeature extends Feature[TweetCandidate, Seq[Long]]

object ConversationModuleFocalTweetIdFeature extends Feature[TweetCandidate, Option[Long]]

object InReplyToFeature extends Feature[TweetCandidate, Option[Long]]

object IsRetweetFeature extends Feature[TweetCandidate, Boolean]

object SourceTweetIdFeature extends Feature[TweetCandidate, Option[Long]]

object SourceUserIdFeature extends Feature[TweetCandidate, Option[Long]]

object SuggestTypeFeature extends Feature[TweetCandidate, Option[SuggestType]]

object ConversationServiceResponseFeatureTransformer

extends CandidateFeatureTransformer[TweetWithConversationMetadata] {

override val identifier: TransformerIdentifier =

TransformerIdentifier("ConversationServiceResponse")

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

Set(

AuthorIdFeature,

InReplyToFeature,

IsRetweetFeature,

SourceTweetIdFeature,

SourceUserIdFeature,

ConversationModuleFocalTweetIdFeature,

AncestorIdsFeature,

SuggestTypeFeature

)

override def transform(candidate: TweetWithConversationMetadata): FeatureMap = {

FeatureMapBuilder()

.add(AuthorIdFeature, candidate.userId)

.add(InReplyToFeature, candidate.inReplyToTweetId)

.add(IsRetweetFeature, candidate.sourceTweetId.isDefined)

.add(SourceTweetIdFeature, candidate.sourceTweetId)

.add(SourceUserIdFeature, candidate.sourceUserId)

.add(ConversationModuleFocalTweetIdFeature, candidate.conversationId)

.add(AncestorIdsFeature, candidate.ancestors.map(\_.tweetId))

.add(SuggestTypeFeature, Some(SuggestType.OrganicConversation))

.build()

}

}