package com.twitter.visibility.builder.tweets

import com.twitter.finagle.stats.StatsReceiver

import com.twitter.notificationservice.model.notification.\_

import com.twitter.servo.util.Gate

import com.twitter.stitch.Stitch

import com.twitter.tweetypie.thriftscala.SettingsUnmentions

import com.twitter.visibility.builder.FeatureMapBuilder

import com.twitter.visibility.common.TweetSource

import com.twitter.visibility.features.NotificationIsOnUnmentionedViewer

object UnmentionNotificationFeatures {

def ForNonUnmentionNotificationFeatures: FeatureMapBuilder => FeatureMapBuilder = {

\_.withConstantFeature(NotificationIsOnUnmentionedViewer, false)

}

}

class UnmentionNotificationFeatures(

tweetSource: TweetSource,

enableUnmentionHydration: Gate[Long],

statsReceiver: StatsReceiver) {

private[this] val scopedStatsReceiver =

statsReceiver.scope("unmention\_notification\_features")

private[this] val requestsCounter = scopedStatsReceiver.counter("requests")

private[this] val hydrationsCounter = scopedStatsReceiver.counter("hydrations")

private[this] val notificationsUnmentionUserCounter =

scopedStatsReceiver

.scope(NotificationIsOnUnmentionedViewer.name).counter("unmentioned\_users")

def forNotification(notification: Notification): FeatureMapBuilder => FeatureMapBuilder = {

requestsCounter.incr()

val isUnmentionNotification = tweetId(notification) match {

case Some(tweetId) if enableUnmentionHydration(notification.target) =>

hydrationsCounter.incr()

tweetSource

.getTweet(tweetId)

.map {

case Some(tweet) =>

tweet.settingsUnmentions match {

case Some(SettingsUnmentions(Some(unmentionedUserIds))) =>

if (unmentionedUserIds.contains(notification.target)) {

notificationsUnmentionUserCounter.incr()

true

} else {

false

}

case \_ => false

}

case \_ => false

}

case \_ => Stitch.False

}

\_.withFeature(NotificationIsOnUnmentionedViewer, isUnmentionNotification)

}

private[this] def tweetId(notification: Notification): Option[Long] = {

notification match {

case n: MentionNotification => Some(n.mentioningTweetId)

case n: FavoritedMentioningTweetNotification => Some(n.mentioningTweetId)

case n: FavoritedReplyToYourTweetNotification => Some(n.replyTweetId)

case n: MentionQuoteNotification => Some(n.mentioningTweetId)

case n: ReactionMentioningTweetNotification => Some(n.mentioningTweetId)

case n: ReplyNotification => Some(n.replyingTweetId)

case n: RetweetedMentionNotification => Some(n.mentioningTweetId)

case n: RetweetedReplyToYourTweetNotification => Some(n.replyTweetId)

case n: ReplyToConversationNotification => Some(n.replyingTweetId)

case n: ReactionReplyToYourTweetNotification => Some(n.replyTweetId)

case \_ => None

}

}

}