package com.twitter.frigate.pushservice.predicate

import com.twitter.finagle.stats.StatsReceiver

import com.twitter.frigate.common.base.TweetAuthorDetails

import com.twitter.frigate.common.base.TweetCandidate

import com.twitter.frigate.common.base.TweetDetails

import com.twitter.frigate.common.predicate.tweet.\_

import com.twitter.frigate.pushservice.model.PushTypes.PushCandidate

import com.twitter.hermit.predicate.NamedPredicate

class LoggedOutPreRankingPredicatesBuilder(implicit statsReceiver: StatsReceiver) {

private val TweetPredicates = List[NamedPredicate[PushCandidate]](

TweetObjectExistsPredicate[

TweetCandidate with TweetDetails

].applyOnlyToTweetCandidatesWithTweetDetails

.withName("tweet\_object\_exists"),

PredicatesForCandidate.oldTweetRecsPredicate.applyOnlyToTweetCandidateWithTargetAndABDeciderAndMaxTweetAge

.withName("old\_tweet"),

PredicatesForCandidate.tweetIsNotAreply.applyOnlyToTweetCandidateWithoutSocialContextWithTweetDetails

.withName("tweet\_candidate\_not\_a\_reply"),

TweetAuthorPredicates

.recTweetAuthorUnsuitable[TweetCandidate with TweetAuthorDetails]

.applyOnlyToTweetCandidateWithTweetAuthorDetails

.withName("tweet\_author\_unsuitable")

)

final def build(): List[NamedPredicate[PushCandidate]] = {

TweetPredicates

}

}

object LoggedOutPreRankingPredicates {

def apply(statsReceiver: StatsReceiver): List[NamedPredicate[PushCandidate]] =

new LoggedOutPreRankingPredicatesBuilder()(statsReceiver).build()

}